

Robert F. Kramer (SBN 181706)  
rkramer@feinday.com  
M. Elizabeth Day (SBN 177125)  
eday@feinday.com  
David Alberti (SBN 220625)  
dalberti@feinday.com  
Sal Lim (SBN 211836)  
slim@feinday.com  
Russell Tonkovich (SBN 233280)  
rtonkovich@feinday.com  
Marc Belloli (SBN 244290)  
mbelloli@feinday.com

**FEINBERG DAY KRAMER ALBERTI  
LIM TONKOVICH & BELLOLI LLP**

577 Airport Boulevard, Suite 250  
Burlingame, California 94010  
Telephone: (650) 825-4300  
Facsimile: (650) 460-8443

*Attorneys for Plaintiff*  
Express Mobile, Inc.

**UNITED STATES DISTRICT COURT  
NORTHERN DISTRICT OF CALIFORNIA  
SAN FRANCISCO DIVISION**

EXPRESS MOBILE, INC.,  
  
*Plaintiff,*  
  
v.  
  
SLACK TECHNOLOGIES, INC.,  
  
*Defendant.*

Case No. 3:21-cv-02001

**COMPLAINT FOR PATENT  
INFRINGEMENT**

**DEMAND FOR JURY TRIAL**

**COMPLAINT FOR PATENT INFRINGEMENT**

1  
2 1. Plaintiff Express Mobile, Inc. (“Express Mobile” or “Plaintiff”), by its attorneys,  
3 demands a trial by jury on all issues so triable and for its Complaint against Slack Technologies,  
4 Inc. (“Slack” or “Defendant”) alleges the following:

**NATURE OF THE ACTION**

5  
6 2. This action arises under 35 U.S.C. § 271 for Slack’s infringement of Express  
7 Mobile’s United States Patent Nos. 6,546,397 (“the ’397 patent”), 9,928,044 (“the ’044  
8 patent”), 9,471,287 (“the ’287 patent”) and 9,063,755 (“the ’755 patent”).

**THE PARTIES**

9  
10 3. Plaintiff Express Mobile, Inc. is an inventor-owned corporation organized under  
11 the laws of the State of Delaware with a place of business at 38 Washington Street, Novato,  
12 California 94947.

13 4. Upon information and belief, Slack Technologies, Inc. is a corporation organized  
14 and existing under the laws of the State of Delaware, with a place of business at 500 Howard  
15 Street, San Francisco, California 94105. Slack may be served through its registered agent for  
16 service at CSC Lawyers Incorporating Service 2710 Gateway Oaks Drive, Suite 150N,  
17 Sacramento, California, 95833.

18 5. Upon information and belief, Slack claims to be the collaboration hub that brings  
19 the right people, information, and tools together to get work done. From Fortune 100 companies  
20 to corner markets, millions of people around the world use Slack to connect their teams, unify  
21 their systems, and drive their business forward.

**JURISDICTION AND VENUE**

22  
23 6. This Court has jurisdiction over the subject matter of this action pursuant to 28  
24 U.S.C. §§ 1331 and 1338(a).

25 7. Upon information and belief, jurisdiction and venue for this action are proper in  
26 the Northern District of California.

27 8. This Court has personal jurisdiction over Slack because Slack has purposefully  
28 availed itself of the rights and benefits of the laws of this State and this Judicial District. Upon

1 information and belief, Slack resides in the Northern District of California by maintaining a  
2 regular and established place of business at 500 Howard Street, San Francisco, California 94105.

3 9. This Court also has personal jurisdiction over Slack because Slack has done and is  
4 doing substantial business in this Judicial District, both generally and, upon information and  
5 belief, with respect to the allegations in this Complaint, including Slack's one or more acts of  
6 infringement in this Judicial District.

7 10. Venue is proper in this Judicial District under 28 U.S.C. §§ 1391(b) and (c) and §  
8 1400(b). Slack has committed acts of infringement through, for example, performing a method  
9 to allow users to produce Internet websites in the Northern District of California and has a  
10 regular and established place of business in this District. Slack's office in San Francisco is a  
11 physical place in the District, it is an established location where Slack's business has been  
12 carried out for several years, and Slack publicly advertises its presence in the District.

### 13 **INTRADISTRICT ASSIGNMENTS**

14 11. Pursuant to Local Rule 3-2 (c), this case involves intellectual property rights and  
15 is subject to assignment on a district wide basis.

### 16 **PATENTS IN SUIT**

17 12. Express Mobile is the lawful owner of all rights, title, and interest in the '397 patent  
18 titled "Browser Based Web Site Generation Tool and Run Time Engine," including the right to  
19 sue and to recover for infringement thereof. The '397 patent was duly and legally issued on April  
20 8, 2003, naming Steven H. Rempell as the inventor. A true and correct copy of the '397 patent is  
21 attached as Exhibit A.

22 13. The inventions of the '397 patent solve technical problems related to website  
23 creation and generation. For example, the inventions enable the creation of websites through  
24 browser-based visual editing tools such as selectable settings panels which describe website  
25 elements, with one or more settings corresponding to commands. These features are exclusively  
26 implemented utilizing computer technology including a virtual machine.

27 14. The claims of the '397 patent do not merely recite the performance of some pre-  
28 Internet business practice on the Internet. Instead, the claims of the '397 patent recite inventive

1 concepts that are rooted in computerized website creation technology, and overcome problems  
2 specifically arising in the realm of computerized website creation technologies.

3 15. The claims of the '397 patent recite inventions that are not merely the routine or  
4 conventional use of website creation systems and methods. Instead, the inventions teach a  
5 browser-based website creation system and method in which the user-selected settings  
6 representing website elements are stored in a database, and in which said stored information is  
7 retrieved to generate said website.

8 16. The technology claimed in the '397 patent does not preempt all ways of using  
9 website or web page authoring tools nor any other well-known prior art technology.

10 17. Accordingly, each claim of the '397 patent recites a combination of elements  
11 sufficient to ensure that the claim amounts to significantly more than a patent on an ineligible  
12 concept.

13 18. In *Express Mobile v. KTree Computer Solutions*, a case filed in the Eastern District  
14 of Texas, the defendant, KTree Computer Solutions, brought a Motion for Judgment on the  
15 Pleadings asserting that the '397 patent is invalid as claiming abstract subject matter under 35  
16 U.S.C. § 101. (C.A. 2:17-00128; Dkt. 9, 17, 22-27). The briefing associated with the motion is  
17 incorporated by reference into this Complaint.

18 19. After considering the respective pleadings, Magistrate Judge Payne recommended  
19 denial of KTree's motion, without prejudice, holding that "the claims appear to address a problem  
20 particular to the internet: dynamically generating websites and displaying web pages based on  
21 stored user-selected settings" and further stating "the asserted claims do not bear all of the  
22 hallmarks of claims that have been invalidated on the pleadings by other courts in the past. For  
23 example, the claims are not merely do-it-on-a-computer claims." (Dkt. 29, attached as Exhibit B.)  
24 No objection was filed to the Magistrate Judge's report and recommendation and the decision  
25 therefore became final.

26 20. In *Express Mobile v. Pantheon Systems, Inc.*, a case filed in the Northern District  
27 of California, the defendant, Pantheon Systems, Inc., brought a Motion to Dismiss Plaintiff's First  
28 Amended Complaint asserting that the '397 patent was directed to the abstract idea of creating and

1 displaying webpages based upon information from a user with no further inventive concept and  
2 purportedly ineligible for patenting under 35 U.S.C. § 101. (Case No. 3:18-CV-04688-RS; Dkt.  
3 26, 32 and 34). The briefing associated with the motion is incorporated by reference into this  
4 Complaint.

5 21. In *Express Mobile v. Code and Theory LLC*, a case filed in the Northern District of  
6 California, the defendant, Code and Theory LLC, brought a Motion to Dismiss Plaintiff's  
7 Complaint asserting that the '397 patent is not subject matter eligible under 35 U.S.C. § 101 as a  
8 matter of law. (Case No. 3:18-CV-04679-RS; Dkt. 35, 40 and 41). The briefing associated with  
9 the motion is incorporated by reference into this Complaint.

10 22. After a hearing and a consideration of the respective pleadings, Hon. Richard  
11 Seeborg denied both motions holding that:

- 12 • "The patents here are directed at a purportedly revolutionary technological  
13 solution to a technological problem—how to create webpages for the internet in a  
14 manner that permits 'what you see is what you get' editing, and a number of other  
15 alleged improvements over the then-existing methodologies." *Id.* at 5.
- 16 • The claims of the '397 patent are "directed to a specific improvement to the way  
17 computers operate," and "it simply cannot be said on the present record that the  
18 claims are drawn so broadly as to be divorced from the potentially patent-eligible  
19 purported technological improvements described in the specification." *Id.* at 5-6.  
20 (Case No. 3:18-CV-04679-RS; Dkt.45; Case No. 3:18-CV-04688-RS Dkt.40;  
21 attached as Exhibit C.)

22 23. In Case Nos. 1:18-CV-01173-RGA and 1:18-CV-01175-RGA, infringement  
23 actions filed by Plaintiff in the District of Delaware, the respective defendants in those actions,  
24 Dreamhost LLC and Hostway Services, Inc., brought Motions to Dismiss claims of the '397  
25 patent on the basis of invalidity under 35 U.S.C. § 101. (Case No. 1:18-CV-01173-RGA D.I. 14,  
26 D.I. 18-21 and 24 Case No. 1:18-CV-01175-RGA D.I. 17-19 and 23). The briefing associated  
27 with the motion is incorporated by reference into this Complaint.  
28

1           24. After consideration of the respective pleadings, Judge Andrews denied both  
2 motions in a joint order, pointing to factual allegations of inventiveness identified by the  
3 Plaintiff, and an expert declaration explaining inventiveness of the claims, noting that such  
4 factual issues preclude a finding of invalidity on a motion to dismiss. (Case No. 1:18-CV-  
5 01173-RGA D.I. 43; Case No. 1:18-CV-01175-RGA D.I. 42; attached as Exhibit D.)

6           25. Plaintiff is the lawful owner of all rights, title, and interest in United States Patent  
7 No. 9,928,044 titled “Systems and Methods for Programming Mobile Devices,” including the  
8 right to sue and to recover for infringement thereof. The ’044 patent was duly and legally issued  
9 on March 27, 2018, naming Steven H. Rempell, David Chrobak and Ken Brown as the inventors.  
10 A true and correct copy of the ’044 patent is attached as Exhibit E.

11           26. The inventions of the ’044 patent solve technical problems associated with  
12 methods and systems for displaying content on displays of devices by providing more efficient  
13 ways of generating, storing and retrieving code for displaying content, for example, dynamic  
14 content, uniformly across different kinds of devices. For example, the inventions of the ’044  
15 patent allow a data-efficient and flexible association between a symbolic name with a User  
16 Interface (“UI”) object (*e.g.*, a UI object for a widget) corresponding to a web component of a  
17 web service, that is manually or automatically selected. The symbolic name has a data format  
18 type corresponding to a subclass of UI objects that support the data format type of the symbolic  
19 name and is only available to UI objects that support the data format of the symbolic name.  
20 Information representative of the defined UI object can be stored in a database and subsequently  
21 retrieved from the database to build an application consisting of at least a portion of the database  
22 using a player, which uses the information to generate one or more web pages for display across  
23 different kinds of devices (*e.g.*, PC, mobile or tablet; or different browsers, operating systems  
24 and applications, including for example both native and browser-based applications.)

25           27. The claims of the ’044 patent do not merely recite the performance of some pre-  
26 Internet business practice on the Internet. Instead, the claims of the ’044 patent recite inventive  
27 concepts that are rooted in the computerized, data-efficient definition, selection, storage and  
28 generation of user defined object attributes (*e.g.*, a UI object for a widget) on displays for

1 different types of devices, such as PC, mobile or tablet or different browsers, and applications.  
2 Such features are specifically grounded in, and overcome problems with data efficiency and  
3 flexibility specifically arising in, the realm of computerized content generation and display  
4 technologies, and are not well-understood, routine and conventional elements.

5 28. For example, the claimed inventions of the '044 patent recite innovative, technical  
6 improvements that select and associate symbolic names with defined UI objects (*e.g.*, UI objects  
7 for a widget) corresponding to web components of web services based on, for example, data  
8 format type, storing information representative of such settings in a database, and building  
9 applications, which together with players, generate uniform, data-efficient content, such as  
10 dynamic content, for display across different types of devices.

11 29. The technology claimed in the '044 patent does not preempt all ways for the  
12 computerized generation of code for a display of a device nor any other well-known or prior art  
13 technology. For example, the specific, innovative technical improvements do not preempt well-  
14 known methods of generating code for a display of a device by programming in HTML or  
15 JavaScript code.

16 30. Accordingly, each claim of the '044 patent thus recites a combination of elements  
17 sufficient to ensure that the claim amounts to significantly more than a patent on an ineligible  
18 concept.

19 31. Plaintiff is the lawful owner of all rights, title, and interest in United States Patent  
20 No. 9,471,287 titled "Systems and Methods for Integrating Widgets on Mobile Devices,"  
21 including the right to sue and to recover for infringement thereof. The '287 patent was duly and  
22 legally issued on October 18, 2016, naming Steven H. Rempell, David Chrobak and Ken Brown  
23 as the inventors. A true and correct copy of the '287 patent is attached as Exhibit F.

24 32. The inventions of the '287 patent solve technical problems associated with  
25 methods and systems for displaying content on displays of devices by providing more efficient  
26 ways of generating code for uniformly displaying content, for example dynamic content, across  
27 different kinds of devices. For example, the inventions of the '287 patent allow a data-efficient  
28 and flexible association between a symbolic name and a UI object (*e.g.*, a UI object for a widget)

1 corresponding to a web component of a web service, that is defined for presentation on a display  
2 of a device. The defined UI object can be selected by a user of an authoring tool or  
3 automatically selected by a system based on a web component selected by the user. Further, the  
4 symbolic name has a data format type corresponding to a subclass of UI objects that support the  
5 data format type of the symbolic name. A device-independent application including the  
6 symbolic name is then produced and provided to the device together with a device-platform-  
7 dependent player. Such operations provide a user-friendly platform allowing the UI object to be  
8 efficiently defined and uniformly displayed across different kinds of devices (*e.g.*, PC, mobile or  
9 tablet; or different browsers, operating systems, and applications, including for example both  
10 native and browser-based applications).

11 33. The claims of the '287 patent do not recite merely the performance of a known  
12 business practice on the Internet. Instead, the claims of the '287 patent recite inventive concepts  
13 grounded in the computerized, data-efficient definition and generation of object attributes (*e.g.*, a  
14 UI object for a widget) on displays for different types of devices, such as PC, tablet, or mobile  
15 devices, or different browsers and applications. Such features are specifically grounded in, and  
16 overcome problems with data efficiency and flexibility specifically arising in, the realm of  
17 computerized content generation and display technologies, and are not well-understood, routine,  
18 and conventional elements.

19 34. For example, the claimed inventions of the '287 patent recite innovative, technical  
20 improvements that associate symbolic names with UI objects (*e.g.*, UI objects for a widget)  
21 corresponding to web components of web services that are manually or automatically selected,  
22 and defined based on, for example, data format type, and produce device-independent  
23 applications including those symbolic names, together with device-dependent players, to provide  
24 uniform, data-efficient server-based content display across different types of devices.

25 35. The technology claimed in the '287 patent does not preempt all ways for the  
26 computerized generation of code for a display of a device nor any other well-known or prior art  
27 technology. For example, the specific, innovative technical improvements do not preempt well-  
28



1 known methods of generating code for a display of a device by programming in HTML or  
2 JavaScript code.

3 36. Each claim of the '287 patent thus recites a combination of elements sufficient to  
4 ensure that the claim amounts to significantly more than a patent on an ineligible concept.

5 37. Accordingly, each claim of the '287 patent recites a combination of elements  
6 sufficient to ensure that the claim amounts to significantly more than a patent on an ineligible  
7 concept.

8 38. Plaintiff is the lawful owner of all rights, title, and interest in United States Patent  
9 No. 9,063,755 titled "Systems and Methods for Presenting Information on Mobile Devices,"  
10 including the right to sue and to recover for infringement thereof. The '755 patent was duly and  
11 legally issued on June 23, 2015, naming Steven H. Rempell, David Chrobak and Ken Brown as  
12 the inventors. A true and correct copy of the '755 patent is attached as Exhibit G.

13 39. The inventions of the '755 patent utilize inventive concepts to solve technical  
14 problems associated with methods and systems for displaying content on displays of devices,  
15 providing more efficient ways of generating code for uniformly displaying content, for example  
16 dynamic content, across different kinds of devices. For example, the inventions of the '755  
17 patent allow a data-efficient and flexible association between a symbolic name and a UI object  
18 (*e.g.*, a UI object for a widget), corresponding to a web component of a web service, that is  
19 defined for presentation on a display of a device. A device-independent application including  
20 the symbolic name is produced and provided to the device, together with a device-platform-  
21 dependent player.

22 40. The claimed inventions of the '755 patent allow the UI object to be efficiently  
23 displayed across different kinds of devices (*e.g.*, PC, mobile or tablet; or different browsers,  
24 operating systems, and applications, including both native and browser-based applications), as  
25 opposed to, for example, programming directly in HTML or JavaScript code. In turn, a user can  
26 enter an input value to the UI object, and obtain an output value based on a web service  
27 associated with the UI object, the input value and output value also being communicated through  
28 symbolic names to provide an additional level of efficiency.

1           41. The claims of the '755 patent do not recite merely the performance of a known  
2 business practice on the Internet. Instead, the claims of the '755 patent recite inventive concepts  
3 concerning the computerized, data-efficient generation of server-based content (*e.g.*, a UI object  
4 for a widget) on displays for different types of devices, such as PC, tablet, or mobile devices, or  
5 different browsers and applications. For example, the claims of the '755 utilize symbolic name  
6 associations and provide device-independent applications including those symbolic names,  
7 together with device-platform-dependent players, to devices. Further, input values and output  
8 values for the defined content are also communicated as symbolic names. Such features are  
9 specifically grounded in, and overcome problems with data efficiency and flexibility specifically  
10 arising in, the realm of computerized content generation and display technologies, and are not  
11 well-understood, routine, and conventional elements.

12           42. For example, the claimed inventions of the '755 patent recite innovative, technical  
13 improvements that associate symbolic names with defined UI objects (*e.g.*, UI objects for a  
14 widget) corresponding to web components of web services, and produce device-independent  
15 applications including those symbolic names, together with device-platform-dependent players,  
16 to provide uniform, data-efficient content, such as dynamic content, for display across different  
17 types of devices.

18           43. The technology claimed in the '755 patent does not preempt all ways for the  
19 computerized generation of code for a display of a device, nor any other well-known or prior art  
20 technology. For example, the specific, innovative technical improvements claimed in the '755  
21 patent do not preempt well-known methods of generating code for a display of a device by  
22 programming in HTML or JavaScript code.

23           44. Each claim of the '755 patent thus recites a combination of elements sufficient to  
24 ensure that the claim amounts to significantly more than a patent on an ineligible concept.  
25 Accordingly, each claim of the '755 patent recites a combination of elements sufficient to ensure  
26 that the claim amounts to significantly more than a patent on an ineligible concept.  
27  
28

**BACKGROUND**

45. Plaintiff Express Mobile is a leader in the business of developing mobile app and web site design and creation platforms, and has intellectual property including U.S. patents relating to certain tools useful in the field. Express Mobile is managed by individuals with decades of technology and business experience. The Chairman of the Board and CTO of Express Mobile, Steve Rempell, is the inventor of Express Mobile's patent portfolio. Mr. Rempell has over 50 years of experience in technology companies, with much of that work focused on web-based technologies and applications.

46. Defendant Slack is a well-known company that claims to be the collaboration hub that brings the right people, information, and tools together to get work done. From Fortune 100 companies to corner markets, millions of people around the world use Slack to connect their teams, unify their systems, and drive their business forward.

**COUNT I - INFRINGEMENT OF U.S. PATENT NO. 6,546,397**

47. Plaintiff incorporates by reference the allegations contained in paragraphs 1 to 46 above.

48. Slack has performed a method to allow users to produce Internet websites which infringed, either literally or under the doctrine of equivalents, one or more claims of the '397 patent in violation of 35 U.S.C. § 271(a).

49. Upon information and belief, Slack directly infringed at least claim 1 of the '397 patent through Slack's platform, including Slack's web app, desktop apps, mobile apps available for Apple iOS devices and Android devices, and related software and servers, that afforded users access to browser-based website authoring tools to edit user-selected settings and for hosting databases for storing information used to generate such websites (collectively, the "Accused Instrumentalities").

50. Slack has infringed at least claim 1 of the '397 patent by having performed a method to allow users to produce Interest websites on and for computers having a browser and a virtual machine capable of generating displays.

51. The Accused Instrumentalities were provided to users via a webapp in a browser. Users of Slack produced Internet websites. The Accused Instrumentalities were also provided to users via a desktop client that was also based on a website model. The Accused Instrumentalities used technologies such as React and Electron to create their web app and desktop apps from a common set of web compatible source code. Similarly, the Accused Instrumentalities provided a mobile app to users that was also a web-based view of its app.

## The Slack Webapp: A 30,000-foot view

The Slack desktop client\* builds and maintains a data model for the duration of your session. There are many elements to the model, but here are a few:

- Channels (Public, Private, and Direct Messages)
- Members (humans and bots comprising your team)
- Messages (from members, or the server—e.g., join/leave events, profile updates etc.)

*\* The Slack real-time messaging app works in web browsers, and our installable desktop app also uses a web-based view of the same; hence, “Desktop” is an interchangeable term for both.*

Source: <https://slack.engineering/making-slack-faster-by-being-lazy/>

52. The Accused Instrumentalities offered apps to the following operating systems and browsers.

Desktop apps		Mobile apps	
Operating System	Requirements	Operating System	Requirements
<a href="#">MacOS</a>	OS X 10.10 or above	<a href="#">iOS</a>	iOS 12.2 or above
<a href="#">Windows</a>	Windows 7 or above	<a href="#">Android</a>	Android 6.0 or above
<a href="#">Linux</a>	Fedora 28 or above Ubuntu LTS releases 16.04 or above Red Hat Enterprise Linux 7.0 or above		
Web browsers			
Browser		Requirements	
Chrome		Version 66 or above	
Firefox		Version 60 or above	
Safari		Version 11.0 or above	
Microsoft Edge		Version 41 or above	

Source: <https://slack.com/help/articles/115002037526-Minimum-requirements-for-using-Slack>

53. The desktop apps of the Accused Instrumentalities included wrappers for web apps, which used technologies such as WebView and Electron.

Originally, the Slack desktop application was written using the [MacGap v1](#) framework, which internally used [WebView](#) to host web content inside of a native app frame. While that served us well for a long time (including the retrofitting of multiple-team support), this architecture was starting to show its age. New features such as HTTP/2 are only coming to Apple's new [WKWebView](#) view, and moving to this would effectively require a complete rewrite of the application. Furthermore, WebView was tied to the operating system's version of Safari, meaning that we didn't have many options when older versions of macOS had an issue in Safari that affected our app.

*Separately, when we created the Slack Windows application, we couldn't use the existing codebase, so we decided to bet on a brand new platform called Electron.*

We've [written about Electron before](#), but to summarize, Electron is a platform that combines the rendering engine from Chromium and the Node.js runtime and module system.

Source: <https://slack.engineering/building-hybrid-applications-with-electron/>

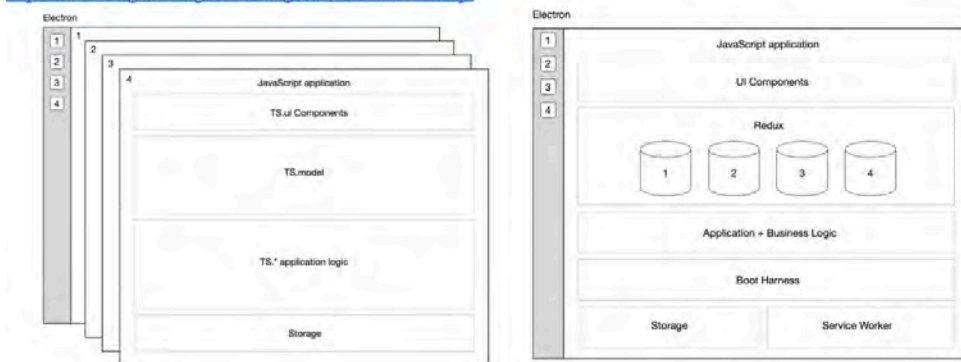
54. At various times, the desktop apps of the Accused Instrumentalities were created with different technologies, such as jQuery, Electron, or React.

Today, after more than half a decade of hyper-growth, Slack is used by millions of people with larger companies working with more data than we ever could have imagined when we first started. Somewhat predictably, a few internal cracks were starting to show in the desktop client's foundation. Additionally, the technology landscape had shifted away from the tools we chose in late 2012 (jQuery, Signals, and direct DOM manipulation) and toward a paradigm of composable interfaces and clean application abstractions. Despite our best efforts to keep things snappy, it became clear that some fundamental changes would be required to evolve the desktop app and prepare it for the next wave of product development.

<https://slack.engineering/rebuilding-slack-on-the-desktop/>

The first order of business was to create the modern codebase. Although this was just a new subdirectory in our codebase, it had three important rules enforced by convention and tooling, each of which was intended to address one of our existing app's shortcomings:

- 1 All UI components had to be built with React
- 2 All data access had to assume a lazily loaded and incomplete data model
- 3 All code had to be "multi-workspace aware"



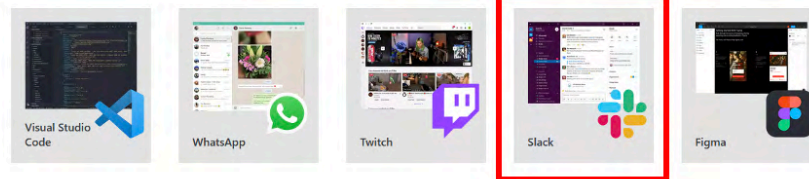
Architecture comparison. New client on the right.

Source: <https://slack.engineering/rebuilding-slack-on-the-desktop/>

See also <https://slack.engineering/growing-pains-migrating-slacks-desktop-app-to-browserview/>



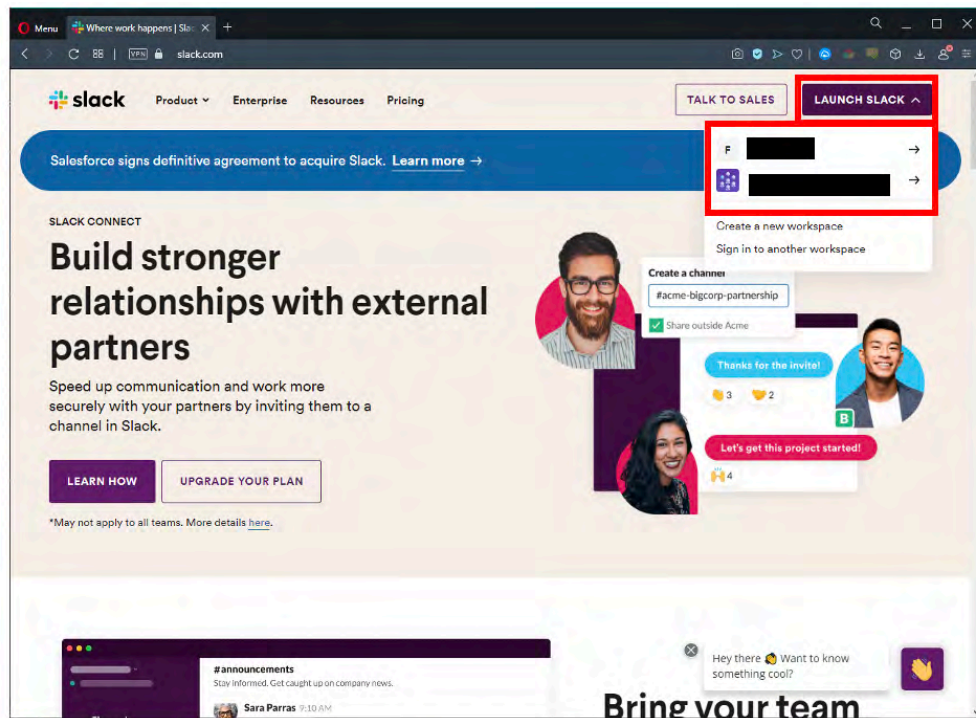
Apps users love, built with Electron



Source: <https://www.electronjs.org/>

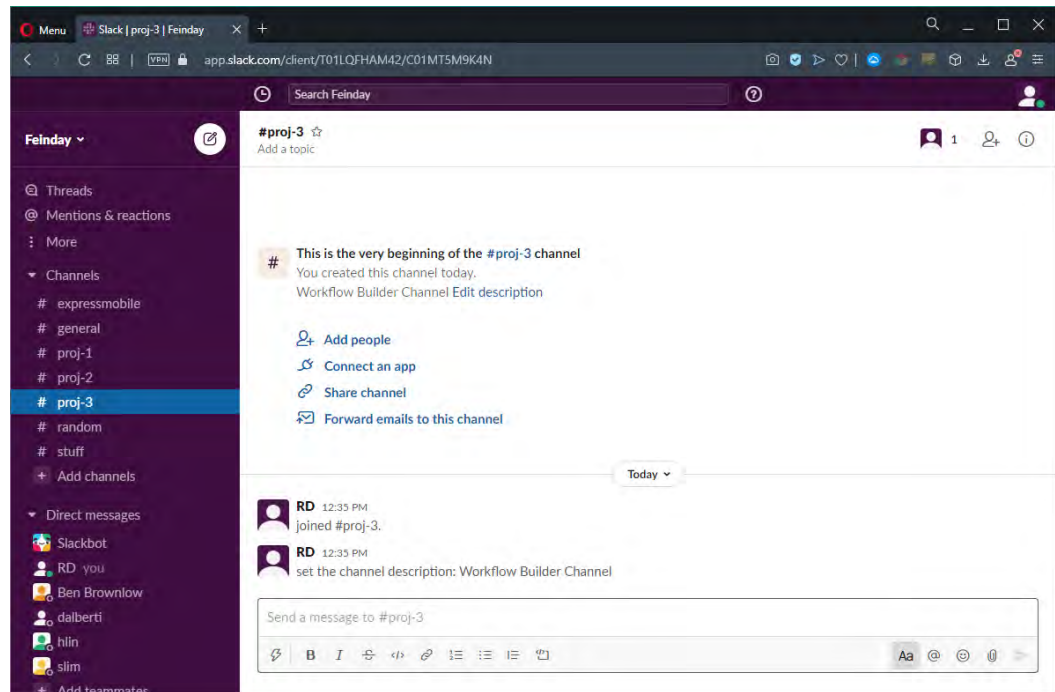
55. The Accused Instrumentalities presented to users an Internet website in their browser displaying a Slack messaging app, through [www.slack.com](http://www.slack.com). If a user was a member of different Slack servers, the user picked a server to join.





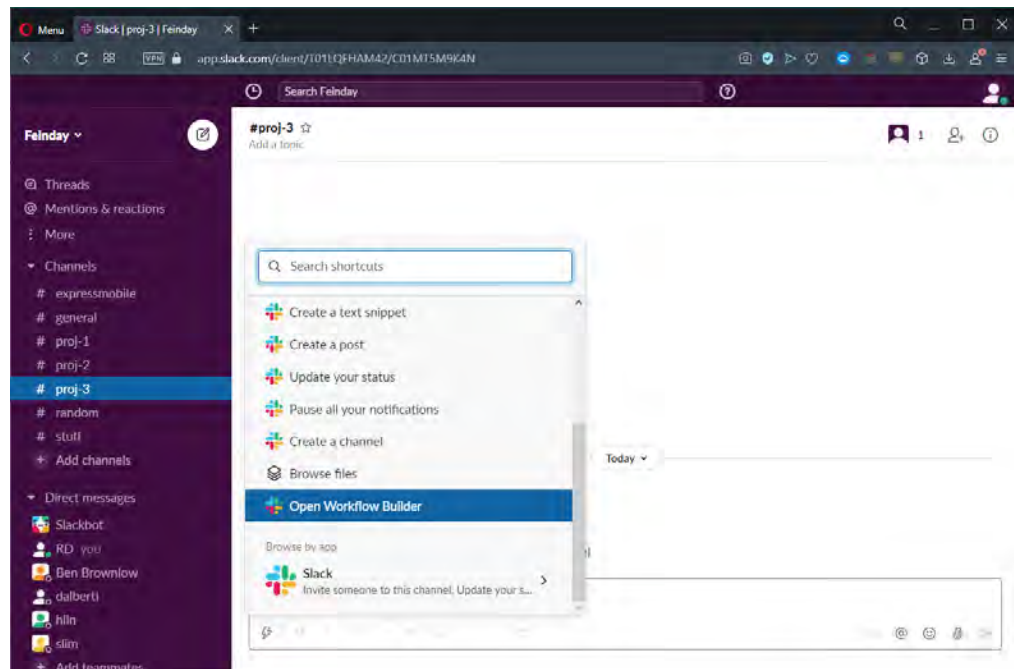
Source: <https://slack.com>

56. The Accused Instrumentalities allowed users to produce Internet websites through the Slack app interface (web app shown below).



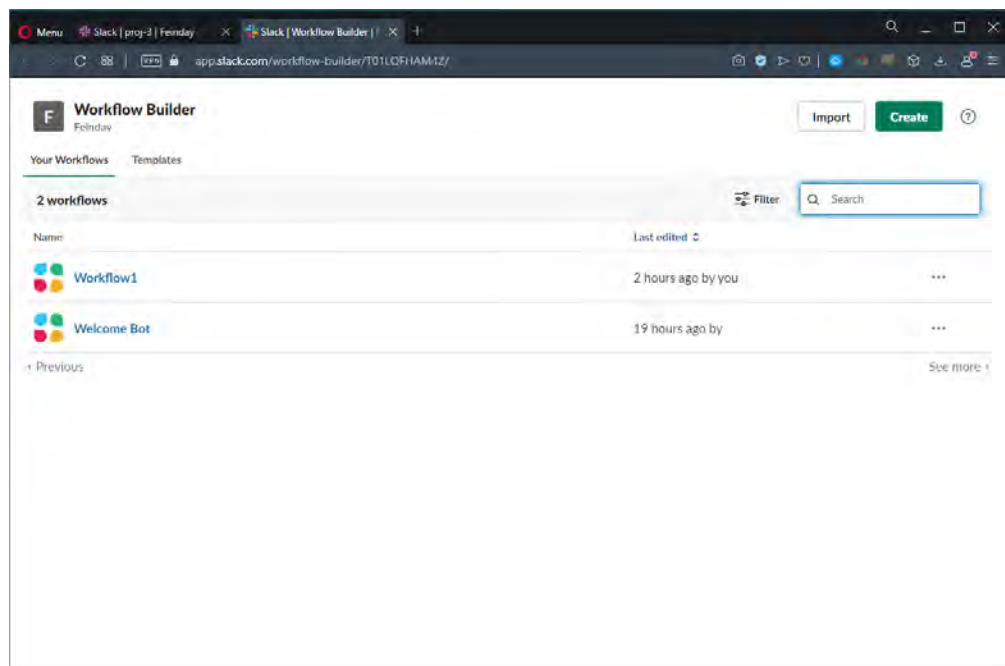
Source: <https://app.slack.com/client/T01LQFHAM42/C01MT5M9K4N>

57. A user could open the Accused Instrumentalities by clicking the lightning bolt, then selecting “Open Workflow Builder.”



Source: <https://app.slack.com/client/T01LQFHAM42/C01MT5M9K4N>

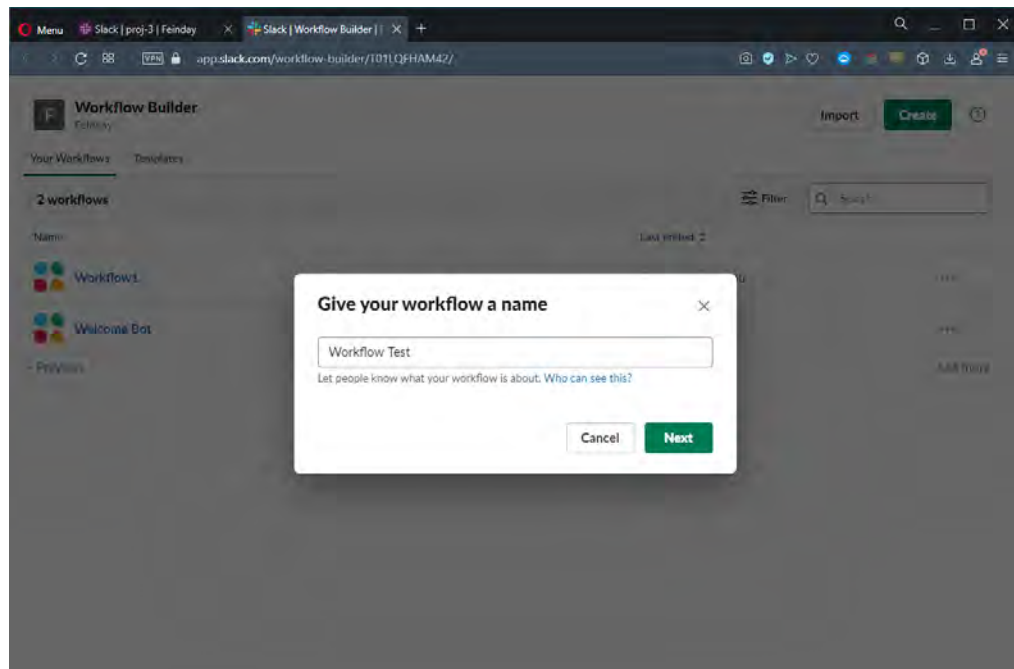
58. The Accused Instrumentalities would open in a new tab.



Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/>

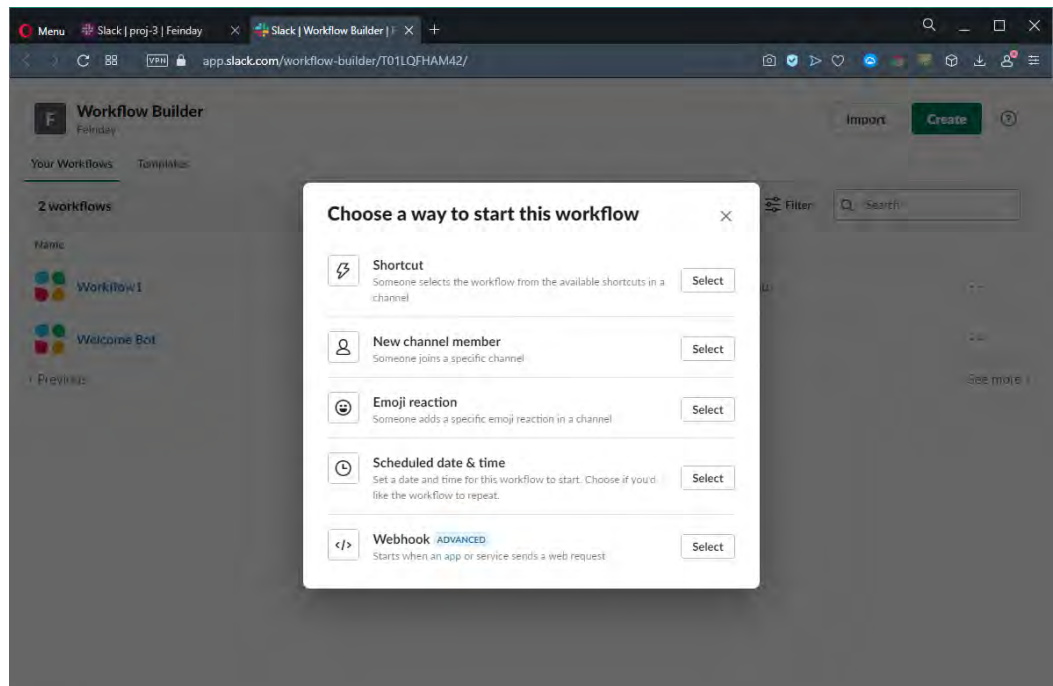


59. After a user clicked “Create,” the Accused Instrumentalities created a new Workflow.

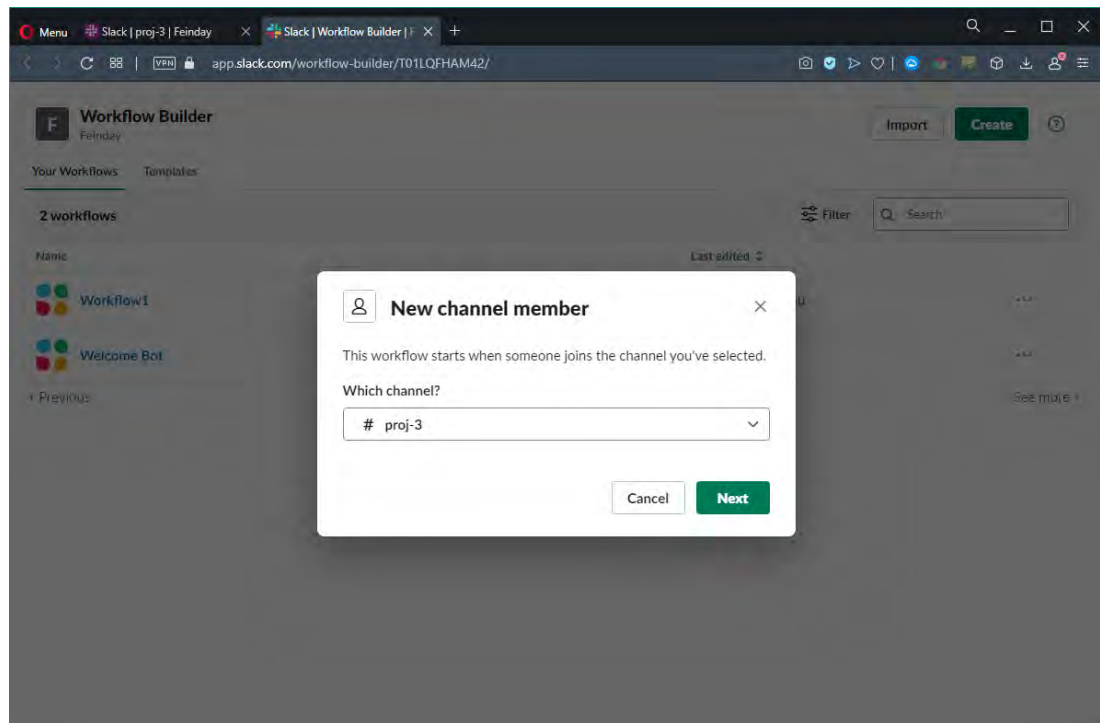


Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/>

60. The Accused Instrumentalities presented the user with several options for a Workflow, for example, “New channel member.”

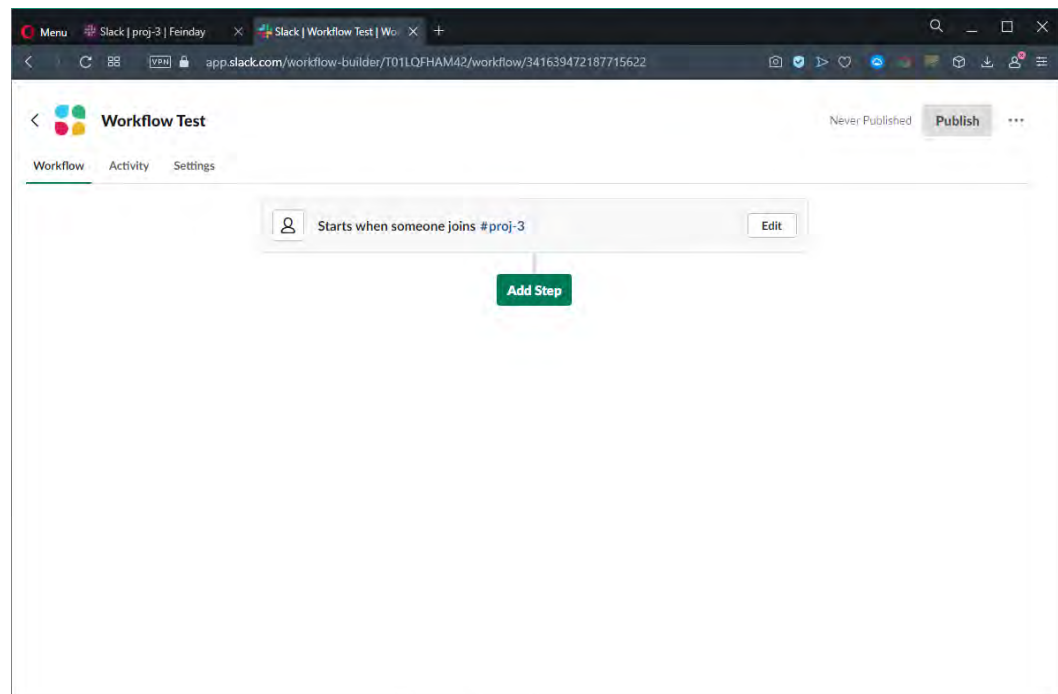


Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/>



Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/>

61. After the user clicked "Next," the Accused Instrumentalities created a Workflow.



Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/workflow/341639472187715622>

62. All modern browsers include an ECMAScript (also known as JavaScript) virtual machine capable of generating displays, by at least modifying the content or properties of a web page.

JavaScript was invented by Brendan Eich in 1995, and became an ECMA standard in 1997. ECMAScript is the official name of the language. From 2015 ECMAScript is named by year (ECMAScript 2015).

## Browser Support

ECMAScript 3 is fully supported in all browsers.

ECMAScript 5 is fully supported in all **modern** browsers.

Source: [https://www.w3schools.com/js/js\\_versions.asp](https://www.w3schools.com/js/js_versions.asp)

## Digging into the JavaScript engine

How does the JavaScript engine fit into this lifecycle? **The JavaScript engine is the component of the browser that takes your JavaScript code, optimizes and executes it.** Also known as virtual machines, JavaScript engines execute your code in an environment that is platform-independent. This means that you can run the same JavaScript code on MacOS, Windows, or Linux. It also means you can run this same code in a browser, like Chrome, or on the desktop, like with Node applications such as **Electron**.

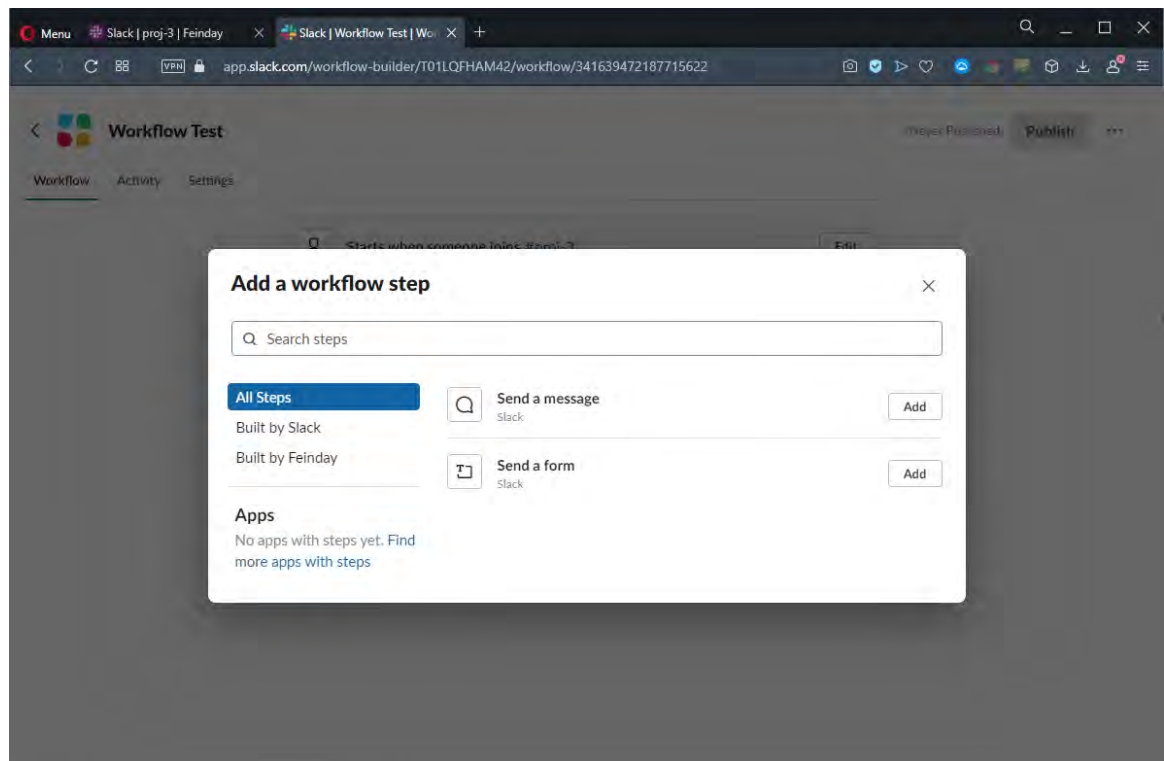
Source: <https://softwareengineeringdaily.com/2018/10/03/javascript-and-the-inner-workings-of-your-browser/>

## How to modify the properties and content of a web page using JavaScript

In this article we will focus on using JavaScript in conjunction with a select element to change the properties of a web page, to show and hide elements in a web page and to modify the content of an element.

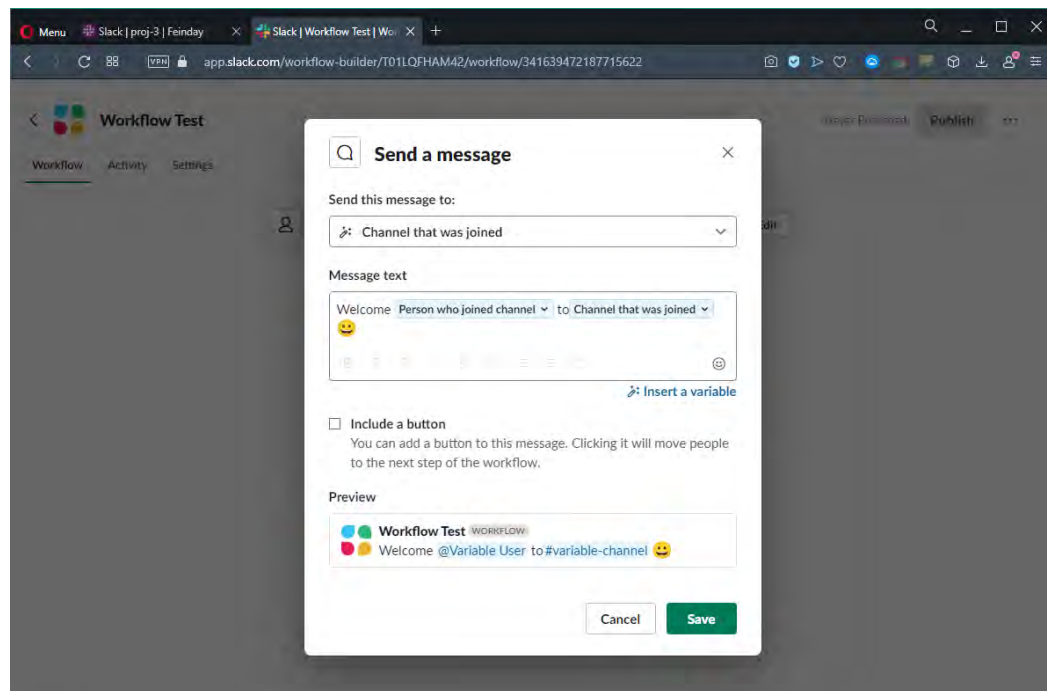
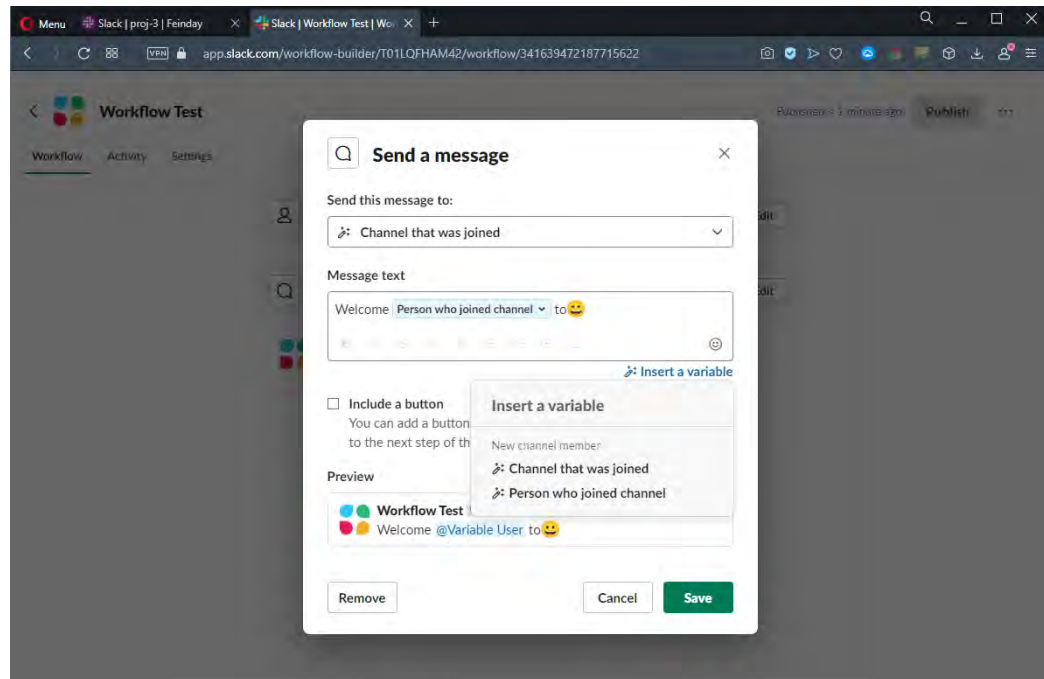
Source: <http://mrbool.com/how-to-modify-the-properties-and-content-of-a-web-page-using-javascript/27684>

63. The Accused Instrumentalities presented viewable menus having a user selectable panel of settings describing elements on a website. These panels of settings were presented through a browser on a computer adapted to accept multiple selectable settings in said panels as inputs therefrom, and these user selectable settings in said panels corresponded to commands to a virtual machine. For example, the Accused Instrumentalities presented a selectable panel of settings for a user to create messages and forms and allowed a user to add a step within the “Workflow Test” to trigger in response to a user joining a channel.



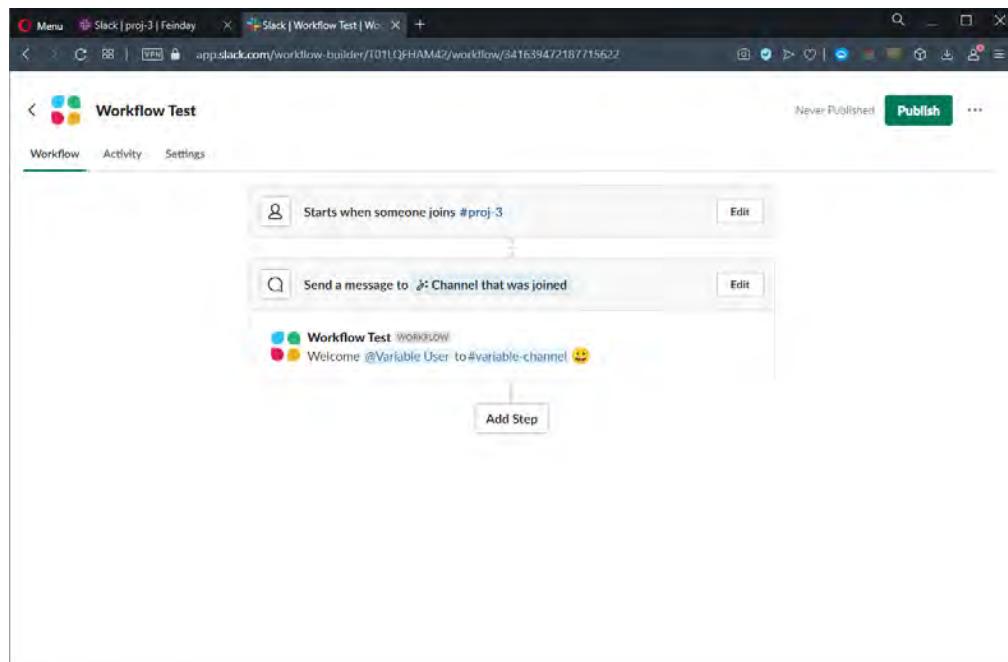
Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/workflow/341639472187715622>

64. In addition, and as another example, in adding a “Send a message” step, the Accused Instrumentalities allowed the user to add a step to send a message to the whole channel when a new user joined. The user could insert formatted text and variables through the Accused Instrumentalities’ window.



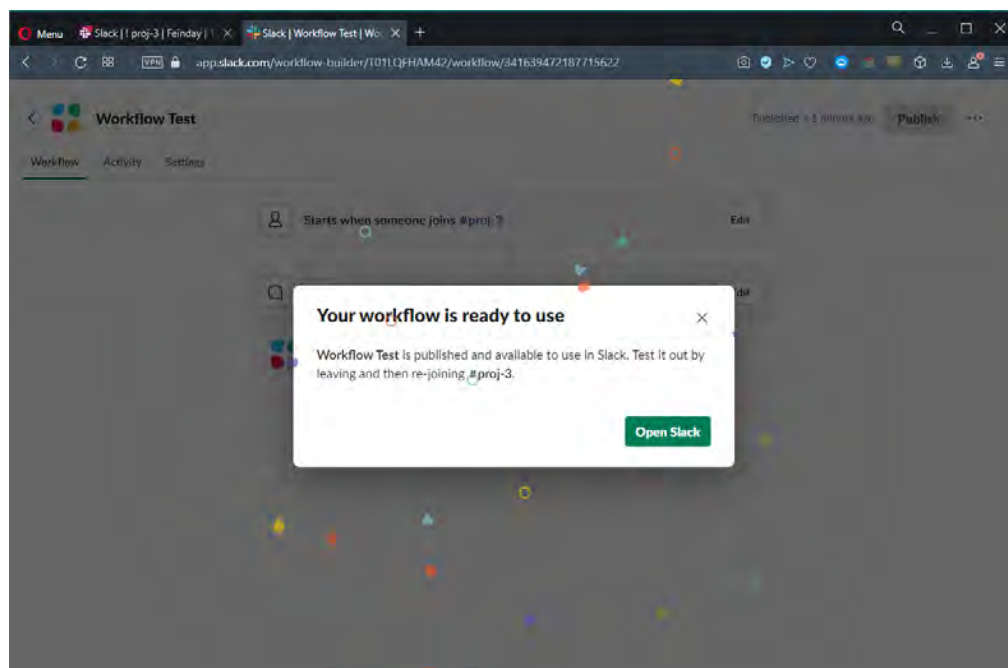
Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/workflow/341639472187715622>

65. The Accused Instrumentalities allowed a user to click “Save” and add the step to the Workflow.



Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/workflow/341639472187715622>

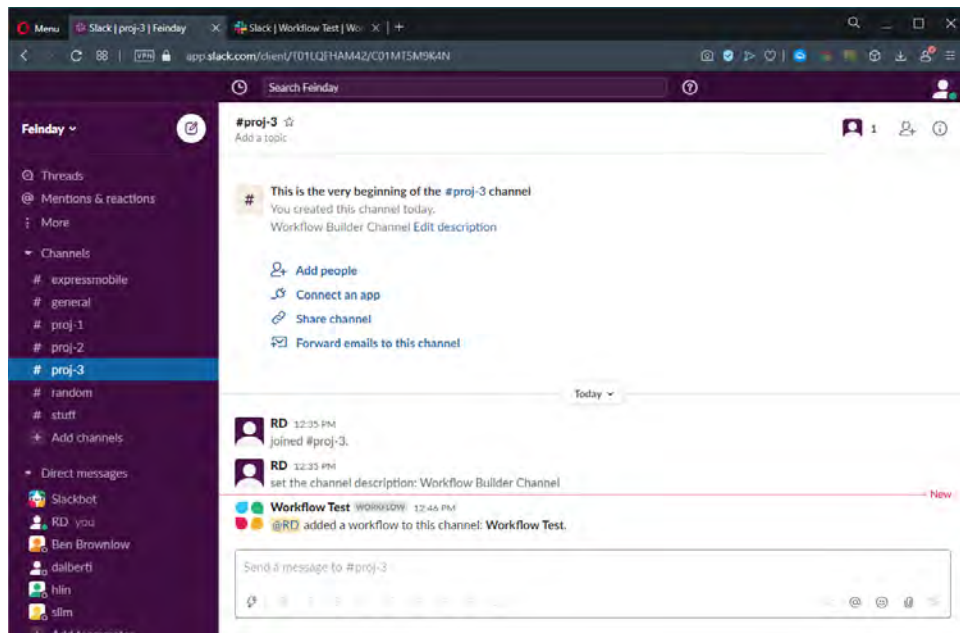
66. The Accused Instrumentalities then published the Workflow.



Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/workflow/341639472187715622>

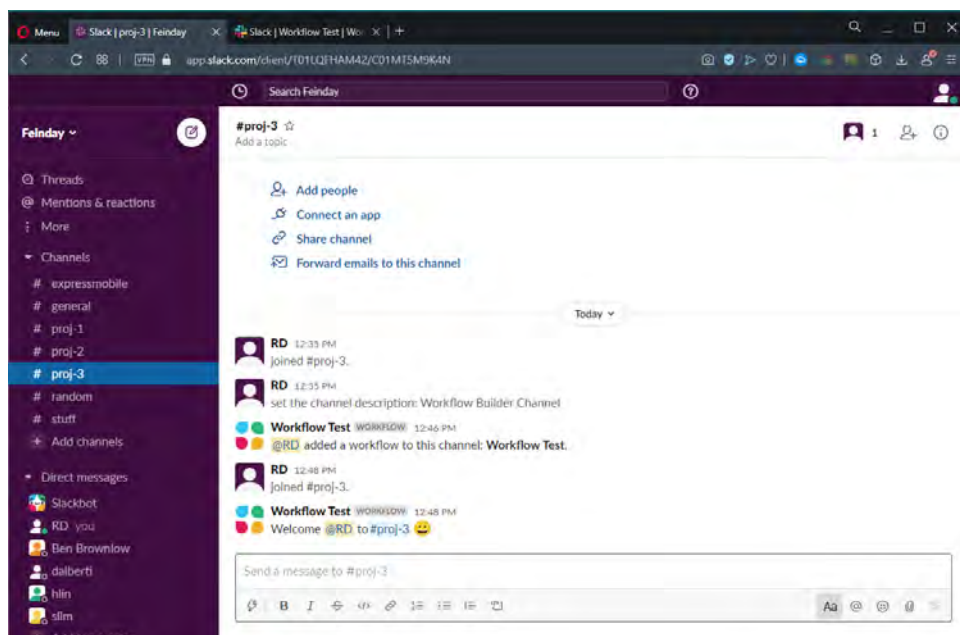


67. The Accused Instrumentalities allowed users in the channel to see that a Workflow was added to the channel.



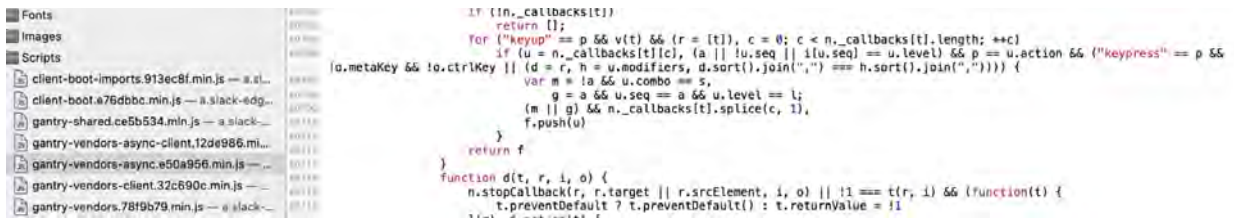
Source: <https://app.slack.com/client/T01LQFHAM42/C01MT5M9K4N>

68. The Accused Instrumentalities sent a message whenever a user left and rejoined a channel.



Source: <https://app.slack.com/client/T01LQFHAM42/C01MT5M9K4N>

69. Examples of user selectable settings describing elements on a website include the font options shown below, such as font styles and formatting settings such as bold, italic, strikethrough, bullets, numbered lists, code snippets, and block quotes. The Accused Instrumentalities included numerous JavaScript code modules including at least those shown below. JavaScript code may be used to process a range of user driven events, including mouse down, keypress, or onload events which may be used to process settings commands sent to the JavaScript virtual machine.



Source: <https://app.slack.com/client/T01LQFHAM42/C01MRQCEV3N>

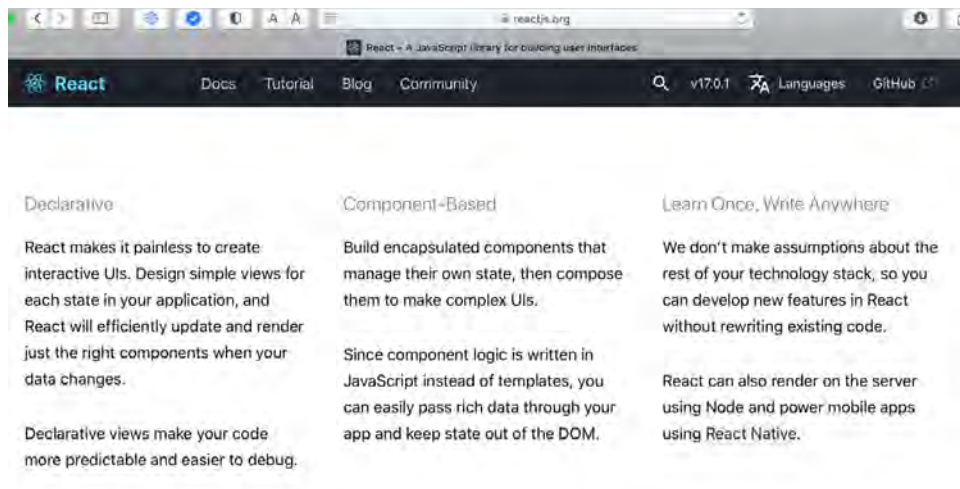
## Mouse Events

onmouseover/onmouseout - When the mouse passes over an element
onmousedown/onmouseup - When pressing/releasing a mouse button
onmousedown - When mouse is clicked: Alert which element
onmousedown - When mouse is clicked: Alert which button
onmousemove/onmouseout - When moving the mouse pointer over/out of an image
onmouseover/onmouseout - When moving the mouse over/out of an image
onmouseover an image map

Source: [https://www.w3schools.com/js/js\\_events\\_examples.asp](https://www.w3schools.com/js/js_events_examples.asp)

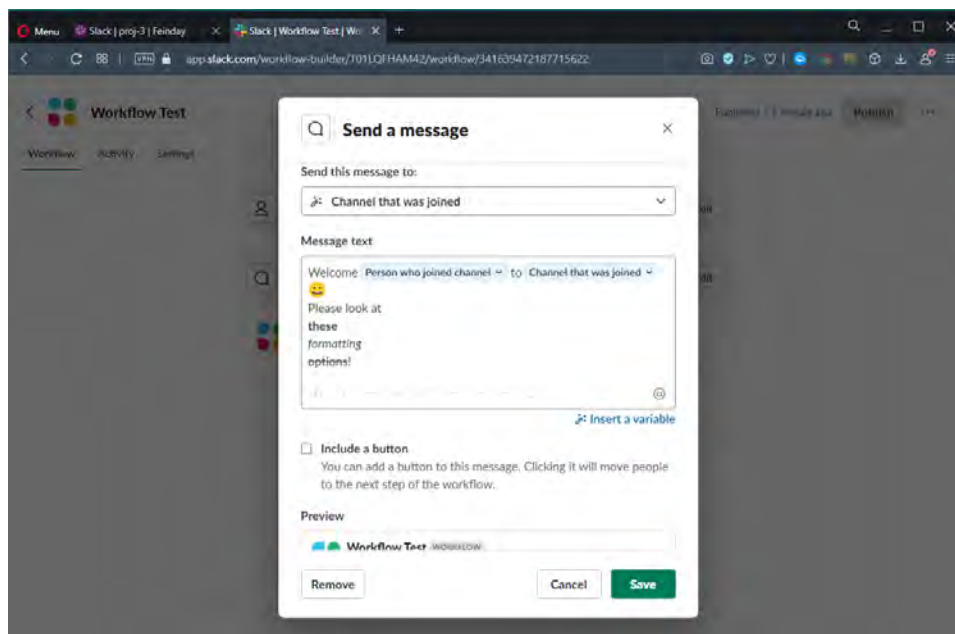
70. As another example, Slack uses React, which is used to support a range of user interface and other events.





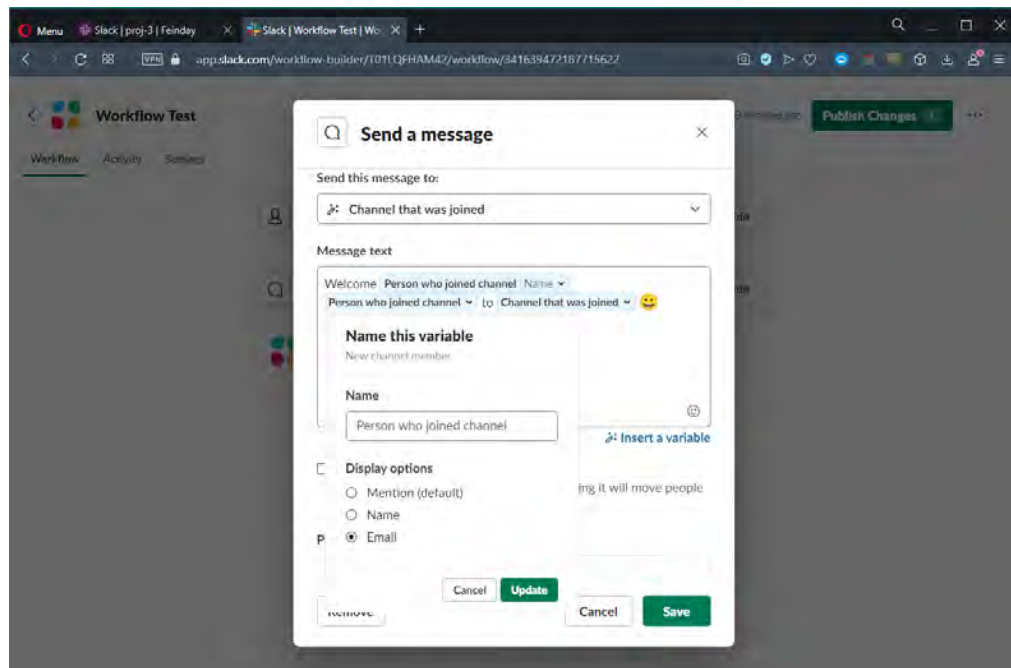
Source: [www.reactjs.org](http://www.reactjs.org)

71. The Accused Instrumentalities generated a display in accordance with one or more user selected settings substantially contemporaneously with the selection thereof. For example, for a text object, a user could have selected formatting settings which controlled the appearance of the text. Upon changing the formatting, the appearance of the text was updated to reflect the new setting. For example, a user could change the formatting of the text below, such as by changing the text to bold, italic, and an ordered list, and the text in the text box would change as settings were selected.



Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/workflow/341639472187715622>

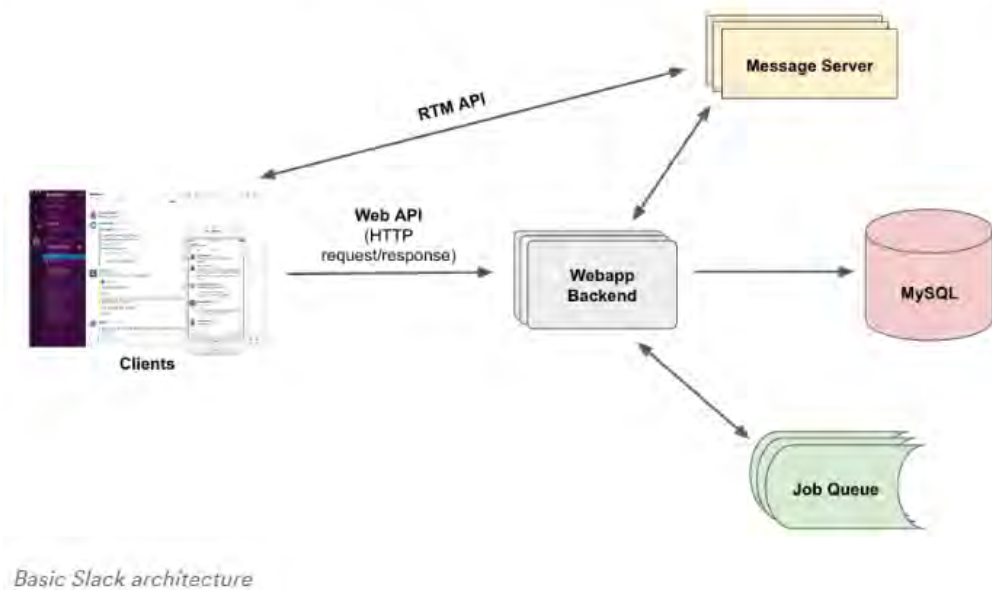
72. The Accused Instrumentalities allowed a user to also change the variables in the message. For example, the user could have added a second “Person who joined channel” and changed the settings of each to “Name” and “Email” respectively, instead of “Mention.”



Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/workflow/341639472187715622>

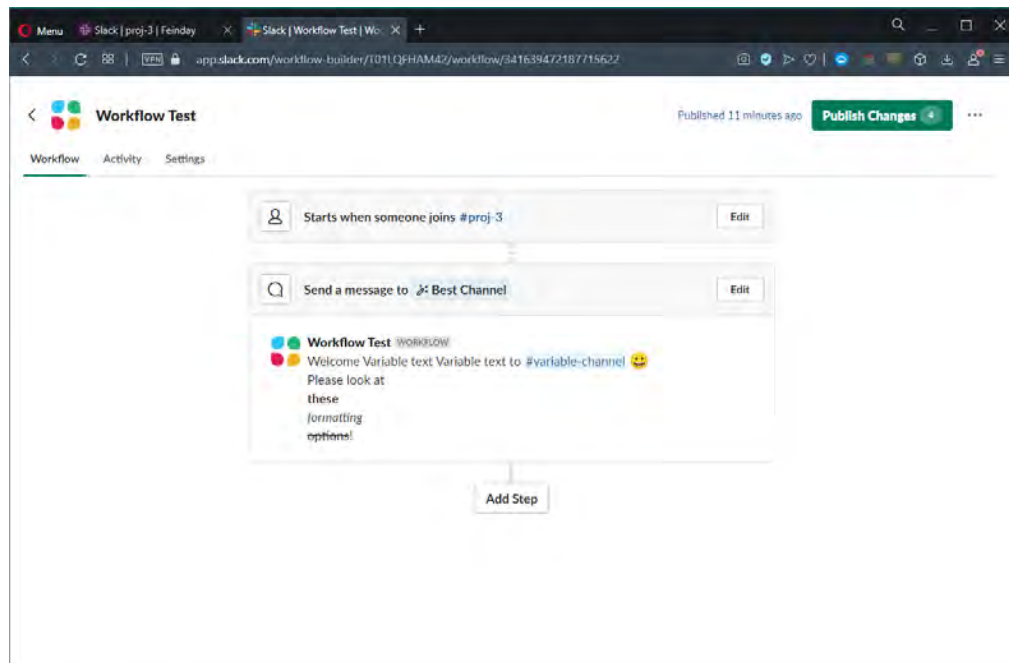
73. The Accused Instrumentalities for the web platform stored information representative of said one or more user selected settings in a database, for example, a MySQL database which contains messages.

For example, a new message posted in a channel is sent via an API call to the webapp servers, where it is sent to the message servers and is persisted in the database. The real-time message servers receive these new messages and send them to connected clients over web sockets, based on who is in the channel.

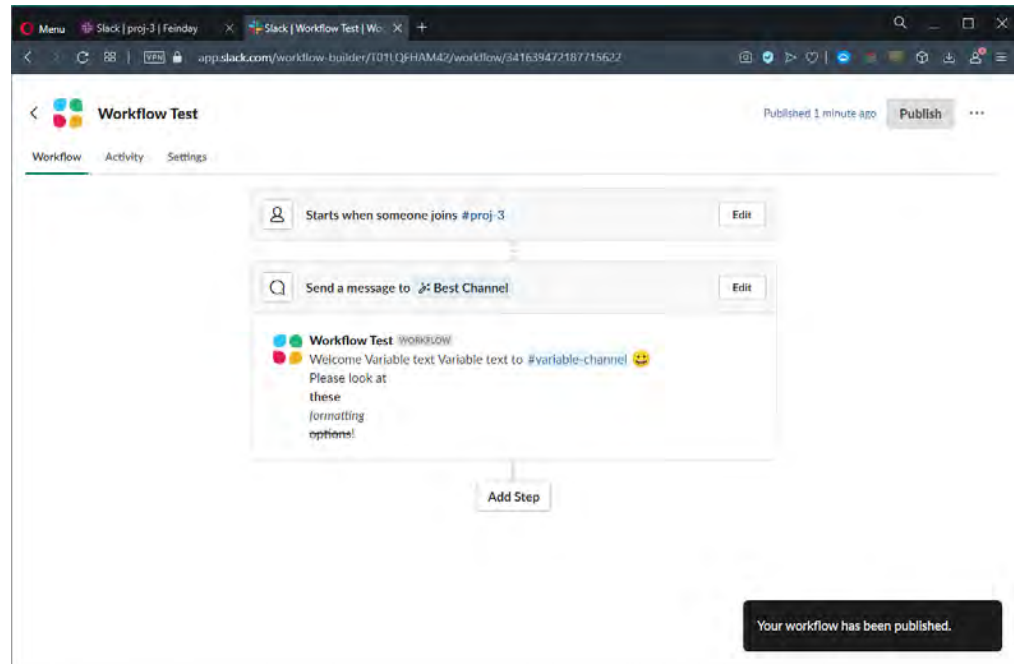


Source: <https://slack.engineering/how-slack-built-shared-channels/>

74. After the user clicked Publish Changes, the Accused Instrumentalities would save the changes to a database.



Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/workflow/341639472187715622>



Source: <https://app.slack.com/workflow-builder/T01LQFHAM42/workflow/341639472187715622>

75. The Accused Instrumentalities generated a web site at least in part by retrieving said information representative of said one or more user selected settings stored in said database. Selections made by the user were used to generate the Slack for the web page displaying the Slack document. For example, the Accused Instrumentalities loaded messages from the database into its servers.

Taking scale into consideration again, we don't want to necessarily pre-load everything; a client could have a lot of channels with unreads (e.g., you return from a week-long vacation). However, it's reasonable to expect that unread Direct Messages (DMs) will be seen as high priority and likely to be read before others, given the 1:1 conversation context. Every unread DM counts as a "mention" and gets badged, so it is highlighted in the client.

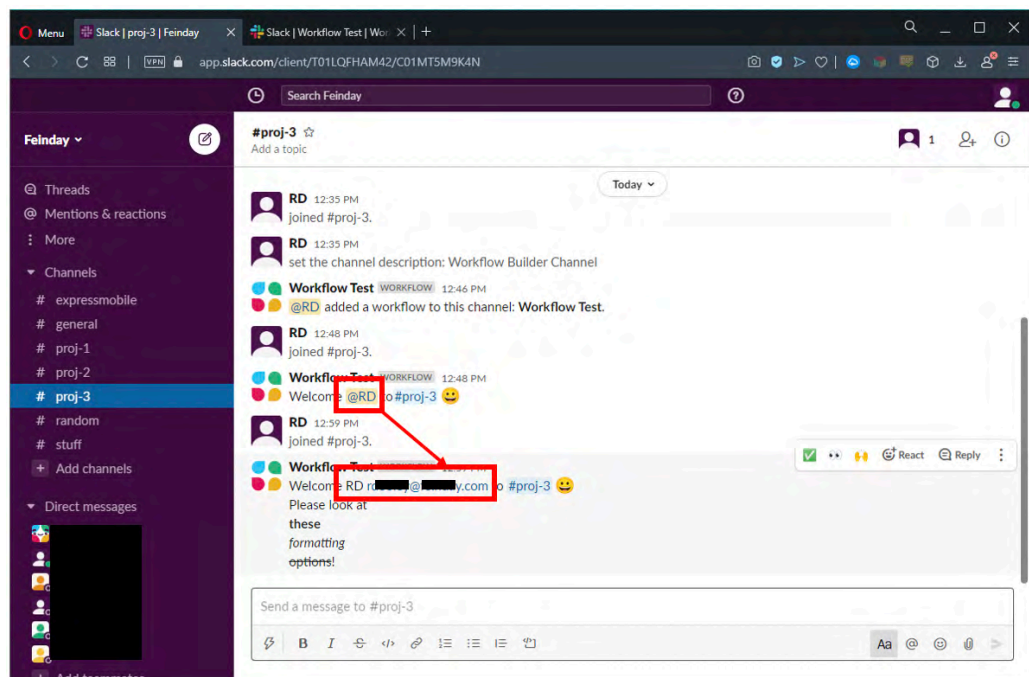
Given that, we prefetch unread messages with the following priority:

- Direct Messages (DMs)
- Group DMs
- Channels with badges (direct mentions or highlight words)
- Channels with unreads, but no badges (up to a limit)

At present, the client will prefetch history for all unread DMs, group DMs, and badged channels because you are likely to be interested in those messages.

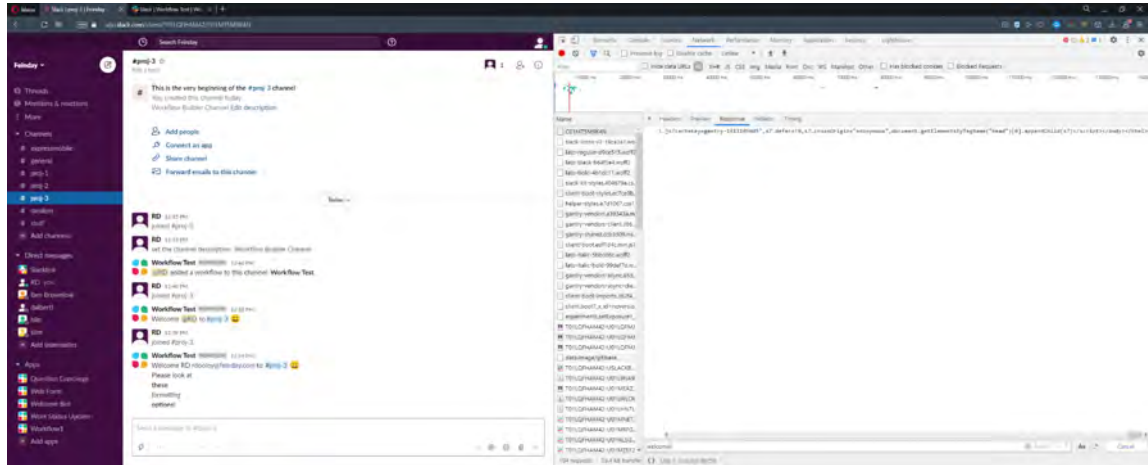
Source: <https://slack.engineering/making-slack-faster-by-being-lazy/>

76. When a user joined the channel, the Accused Instrumentalities would send the new message, reflecting the changes to the settings that the user made and were saved in the database. For example, the formatting changes such as the bold, italic, and strikethrough were reflected in the text, as well as the changes to the variables, which showed the name and email instead of a mention.

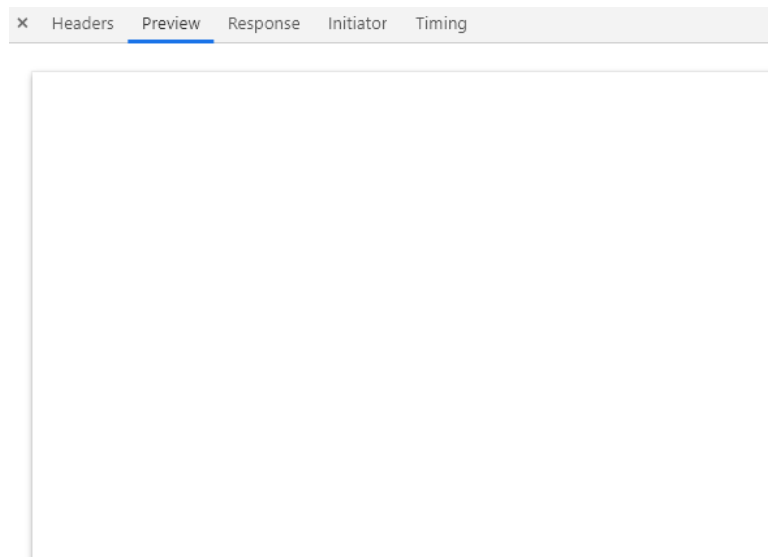


Source: <https://app.slack.com/client/T01LQFHAM42/C01MT5M9K4N>

77. As another example, after the Accused Instrumentalities refreshed the Slack widow, when the web page for the app initially loaded, it was blank with very little HTML for display. For example, the content such as “welcome” did not appear in the original page load.

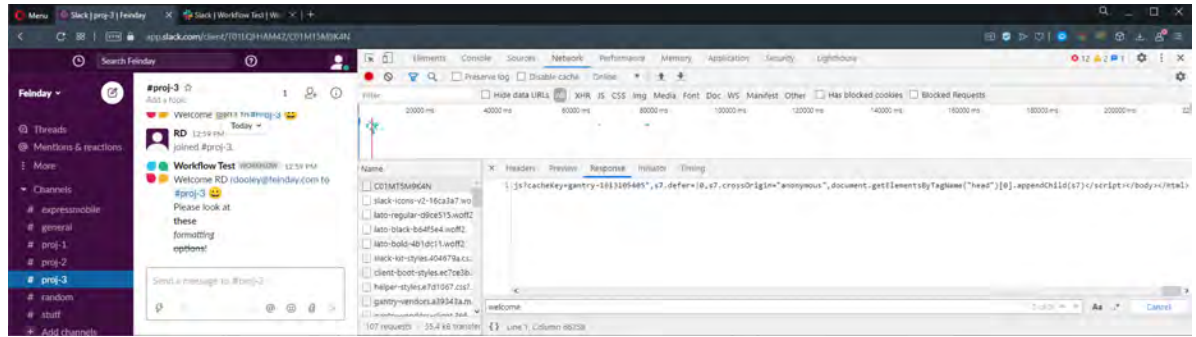


Source: <https://app.slack.com/client/T01LQFHAM42/C01MT5M9K4N>



Source: <https://app.slack.com/client/T01LQFHAM42/C01MT5M9K4N>





Source: <https://app.slack.com/client/T01LQFHAM42/C01MT5M9K4N>

78. The Accused Instrumentalities built the document web page to generate said website from a database, including using at least one run time file which utilized information stored in the database to generate virtual machine commands for the display of at least a portion of the web page. Upon loading of the document web page, the Accused Instrumentalities for the web platform page had very little displayable content, but substantially contained JavaScript code and JavaScript runtime file references used to load a series of JavaScript runtime files and initialize the web page. At least one runtime file loaded the data from a database which was used to generate the document web page. The initial page load shows HTML elements which are empty but accompanied by JavaScript code and commands to load additional JavaScript runtime files. This initial JavaScript code contained at least code to load additional JavaScript files which are used to generate the full document page. This code loaded additional code and other files to generate and display the website, including settings information.

79. Slack's infringement has damaged Express Mobile.

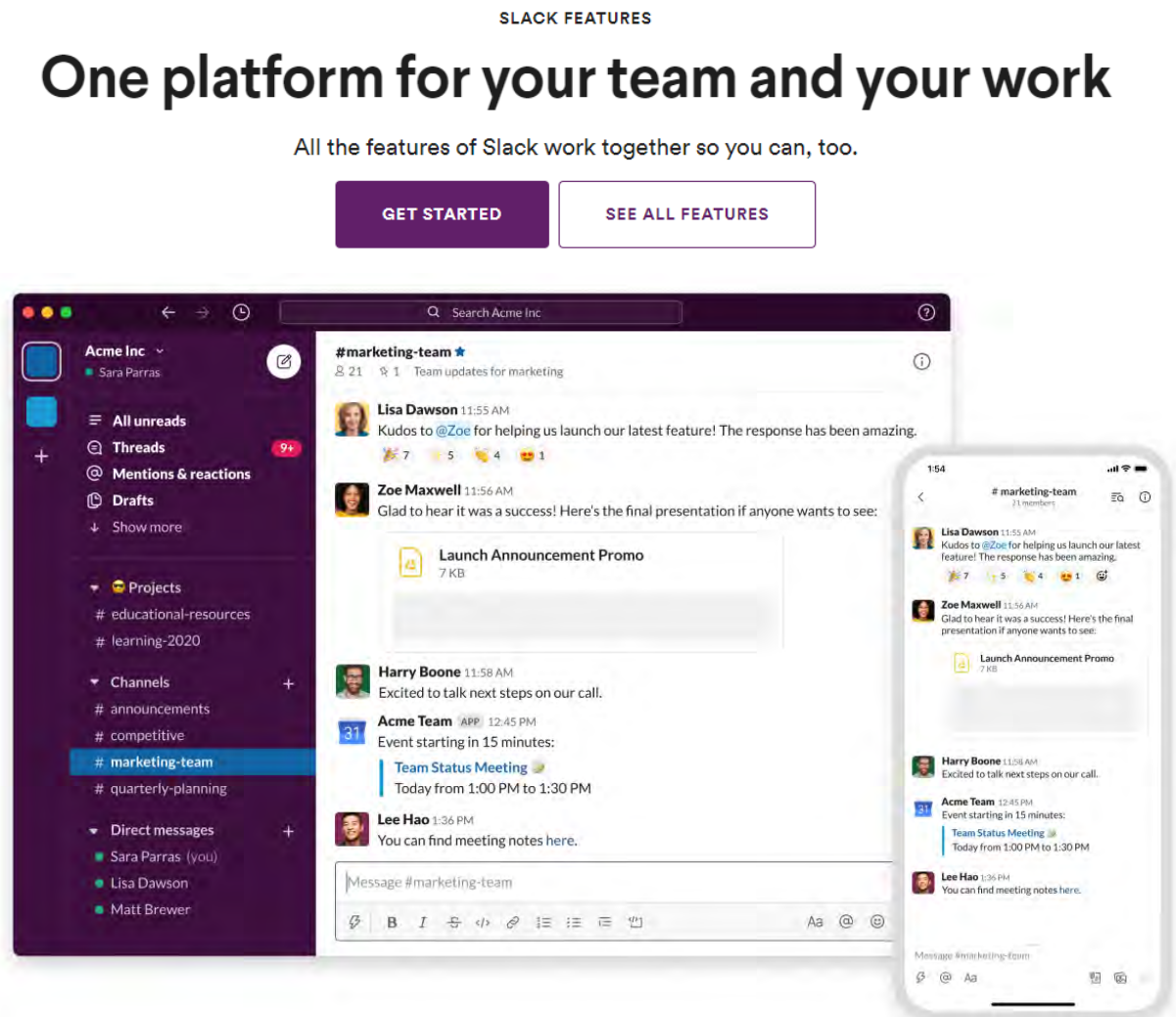
## **COUNT II - INFRINGEMENT OF U.S. PATENT NO. 9,928,044**

80. Plaintiff incorporates by reference the allegations contained in paragraphs 1 to 79 above.

81. Slack has and is making, using, offering for sale, and/or selling (including licensing) in the United States Slack's platform, including Slack.com, Slack Workflow Builder, the Slack mobile application available for Apple iOS devices (including Apple TV, iPads, iPhones and iPod Touch) on Apple's App Store, Android devices (including phones and tablets) on Google Play,

and related software and servers, that afford users access to a system for generating code to provide content on a display of a device (collectively, the “Accused Instrumentalities”) that infringes, either literally or under the doctrine of equivalents, one or more claims of the ’044 patent in violation of 35 U.S.C. § 271(a).

82. Upon information and belief, Slack has directly infringed at least claim 1 of the ’044 patent through the Accused Instrumentalities that comprise a system for generating code to provide content on the display of a device for each of its users generate code to provide content on a display of a device.



Source: <https://slack.com/features>



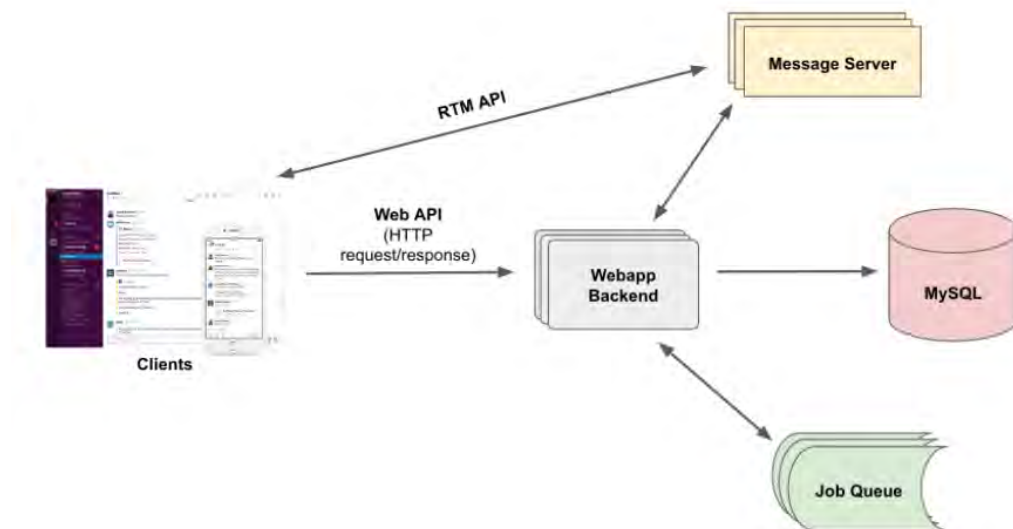
83. The Accused Instrumentalities include computer memory to store user selections and other information relating to a user's web site, web pages, and the web components or elements that are intended to be displayed.

Before we dive into shared channels, let's take a look at how Slack works at a high level. Slack implements a client-server architecture where clients (mobile, desktop, web and apps) talk to two backend systems:

- 1 **1** *webapp servers*, which handle HTTP request/response cycles and communicate with other systems like the main databases and job queues
- 2 **2** *real-time message servers*, which fan out messages, profile changes, user presence updates, and a host of other events to clients

For example, a new message posted in a channel is sent via an API call to the webapp servers, where it is sent to the message servers and is persisted in the database. The real-time message servers receive these new messages and send them to connected clients over web sockets, based on who is in the channel.

Source: <https://slack.engineering/how-slack-built-shared-channels/#:~:text=%20Slack%20implements%20a%20client-server%20architecture%20where%20clients,a%20host%20of%20other%20events%20to...%20More%20>



*Basic Slack architecture*



Source: <https://slack.engineering/how-slack-built-shared-channels/#:~:text=%20Slack%20implements%20a%20client-server%20architecture%20where%20clients,a%20host%20of%20other%20events%20to...%20More%20>

84. The Accused Instrumentalities includes symbolic names required for evoking one or more web components each related to a set of inputs and outputs of a web service obtainable over a network. The symbolic names, such as parameter names, are character strings that do not contain either a persistent address or pointer to an output value accessible to the web service.

85. The Accused Instrumentalities include a feature called Workflow Builder, which offers a set of tools to help users automate processes into workflows (or custom applications) in their Slack workspace. Workflows require the use of a symbolic name, or parameter value, to evoke a web component related to a set of inputs and outputs of a web service obtainable over a network.

## Guide to Workflow Builder

### Who can use this feature?

-  By default, all **members** of a workspace can [create workflows](#). Any **member** or **guest** can use workflows added to channels they belong to.
-  **Standard**, **Plus**, and **Enterprise Grid** plans

[Workflow Builder](#) offers a set of tools — no coding required! — to help you automate routine processes into workflows in your Slack workspace. Workflows in Slack can be as simple or as complex as you'd like, and can even be connected to other apps and services you use to get work done.

Read on to learn about Workflow Builder, then use our tutorials as a guide to build your workflow.

Source: <https://slack.com/help/articles/360035692513-Guide-to-Workflow-Builder>

## Modals

Modals are dedicated windows that allow your app to collect input from users and display dynamic information in a focused space. Modals must be initiated by a user (with features like buttons, slash commands, and other invocative actions).

Source: <https://api.slack.com/tutorials/design-expense-block-kit>

## Workflow Builder glossary

<b>Trigger</b>	How your workflow will start. You can choose a ⚡ <b>shortcuts menu</b> trigger to launch your workflow from the menu in a channel, or another trigger to start your workflow automatically when a certain action is taken.
<b>Step</b>	How people interact with your workflow. You can add steps that send messages or custom forms, or use <a href="#">steps from Slack apps</a> to link your workflow to other tools.
<b>Variable</b>	Information submitted to your workflow that can be referenced in steps. More variables will be available to choose from as you add steps to your workflow.
<b>Collaborator</b>	A member of your workspace with permission to manage your workflow. Any collaborator can change or remove steps, manage other collaborators, and unpublish or delete a workflow.

Source: <https://slack.com/help/articles/360035692513-Guide-to-Workflow-Builder>

## Set up your workflow

While you'll first be presented with picking a name for your workflow, the more important decision you'll need to make is how your workflow will start.

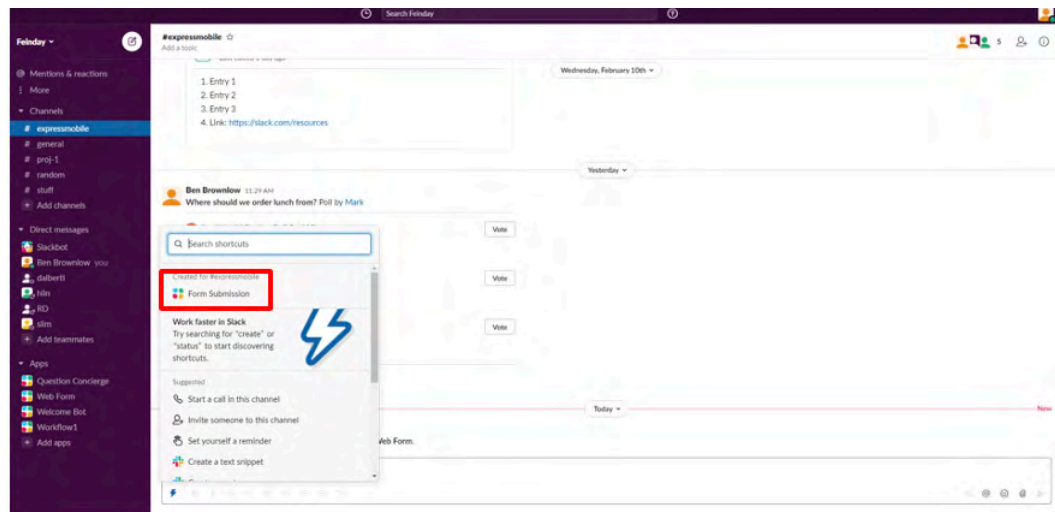
### Choose a trigger

The trigger you pick for your workflow determines how it will start. Workflows are channel-specific, and can only be used in the channel you choose.

Trigger	How it works
⚡ Shortcuts menu	Your workflow starts when someone selects it from the shortcuts menu.
👤 New channel member	Your workflow starts automatically when someone joins the channel.
😊 Emoji reaction	Your workflow starts automatically when someone in the channel adds a certain <a href="#">emoji reaction</a> to a message.
🕒 Scheduled date & time	Your workflow starts automatically at a set date and time.
</> Webhook*	Your workflow starts automatically <a href="#">when a custom webhook sends a web request to Slack</a> from another app or service.

\*Creating workflows with webhooks requires some technical expertise.

Source: <https://slack.com/help/articles/360053571454-Set-up-a-workflow-in-Slack>



86. The Accused Instrumentalities include an address of the web service.

## Network settings

Slack must have a persistent connection between our messaging server and members' apps or browsers. To do so, Slack uses WebSockets over port 443.

Proxies and firewalls can sometimes interrupt this connection. If someone using Slack can't connect from a specific location, or if there's a WebSocket failure in our [Slack connection test](#), you'll need to adjust your proxy or firewall to keep the connection to Slack open. Here's how:

- ✔ Visit <https://my.slack.com/help/urls> and add all specified URLs to your allowlist. If your environment requires access to multiple workspaces and/or orgs, visit that page from each workspace and/or org.
- ✔ Check if your proxy is running SSL decryption. If it is, the proxy must either support WebSockets, or you'll need to exempt the following domains: \*.lb.slack-msgs.com wss-primary.slack.com wss-backup.slack.com wss-mobile.slack.com

Network administrators can restrict access to specific Slack workspaces using [proxy header injection](#) on their network. The header injection solution has the same network requirements as above.

```

1      "*.chime.aws",
2      "*.lb.slack-msgs.com",
3      "a.slack-edge.com",
4      "a.slack-imgs.com",
5      "admin.slack.com",
6      "alpha.slack.com",
7      "api.slack.com",
8      "app.slack.com",
9      "assets.slack.com",
10     "auth.slack.com",
11     "avatars.slack-edge.com",
12     "b.slack-edge.com",
13     "b.slack-imgs.com",
14     "beta.slack.com",
15     "blog.slack.com",
16     "ca.slack-edge.com",
17     "downloads.slack-edge.com",
18     "edgeapi.slack.com",
19     "email.slack.com",
20     "email2.slack.com",
21     "email3.slack.com",
22     "email4.slack.com",
23     "emoji.slack-edge.com",
24     "example.slack.com",
25     "feinday.slack.com",
26     "files-edge.slack.com",
27     "files-origin.slack.com",
28     "files.slack.com",
29     "global-upload-edge.slack.com",
30     "go-beta.slack.com",
31     "go-debug.slack.com",
32     "go.slack.com",
33     "help.slack.com",
34     "hooks.slack.com",
35     "join.slack.com",
36     "my.slack.com",
37     "oauth2.slack.com",
38     "platform-tls-client.slack.com",
39     "platform.slack-edge.com",
40     "slack-email.slack.com",
41     "slack-files.com",
42     "slack-imgs.com",
43     "slack-infra.slack.com",
44     "slack-redirect.com",
45     "slack-redirect.net",
46     "slack.com",
47     "slack.global.ssl.fastly.net",
48     "slackb.com",
49     "spellcheck.slack-edge.com",
50     "status.slack.com",
51     "try.slack.com",
52     "universal-upload-edge.slack.com",
53     "upload.slack.com",
54     "wss-backup.slack.com",
55     "wss-mobile.slack.com",
56     "wss-primary.slack.com"

```

Source: <https://slack.com/help/articles/360001603387-Manage-Slack-connection-issues>

87. The Accused Instrumentalities include an authoring tool configured to define a UI object for presentation on the display. The UI object corresponds to a web component which consists of an input of the web service and an output of the web service. For example, Workflow

Builder allows a user to define a UI object corresponding to a web component for presentation on the display.

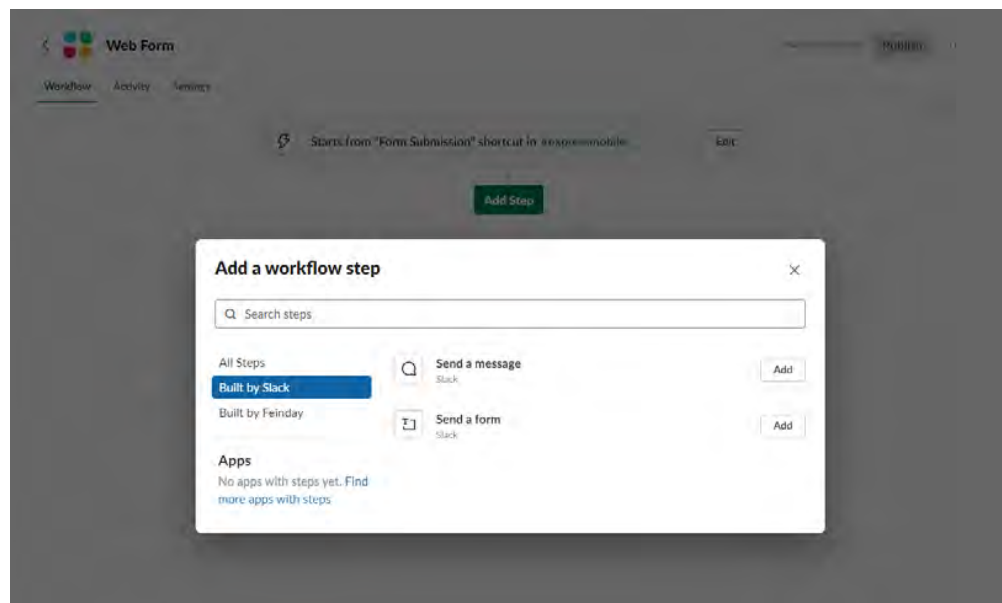
### Form steps

Add **forms** to your workflow to organize and collect information. Forms can include space for custom responses, or prompt people to choose options from a list. Once a form has been completed, you can choose to send responses to a channel or a specific person via DM.



**Note:** Form responses cannot be edited once they've been submitted via Workflow Builder.

Source: <https://slack.com/help/articles/360053571474-Add-steps-to-a-workflow>





**Send a form**

Title: Simple Form

Questions

Add a question

What is your favorite color?

Choose a question type

Select from a list

Short answer

Long answer

✓ Select from a list

Select a person

Select a channel or DM

Default selection

No selection

☒ Make this required

Add Question

☒ Send submitted responses to a channel or to someone in a DM

Cancel Save

**Send a form**

Questions

Add a question

What is your favorite color?

Choose a question type

Select from a list

List Items

Red

Green

Blue

Add List Item

Default selection

No selection

☒ Make this required

Add Question

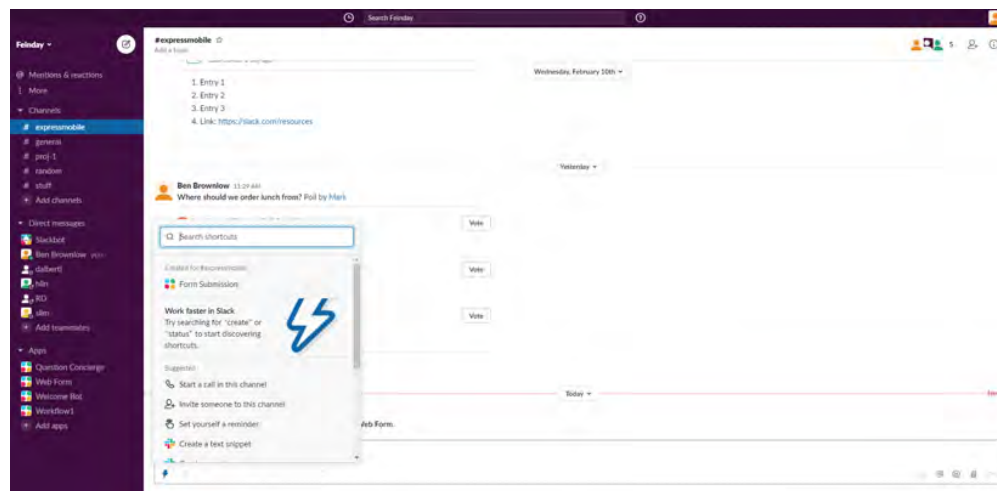
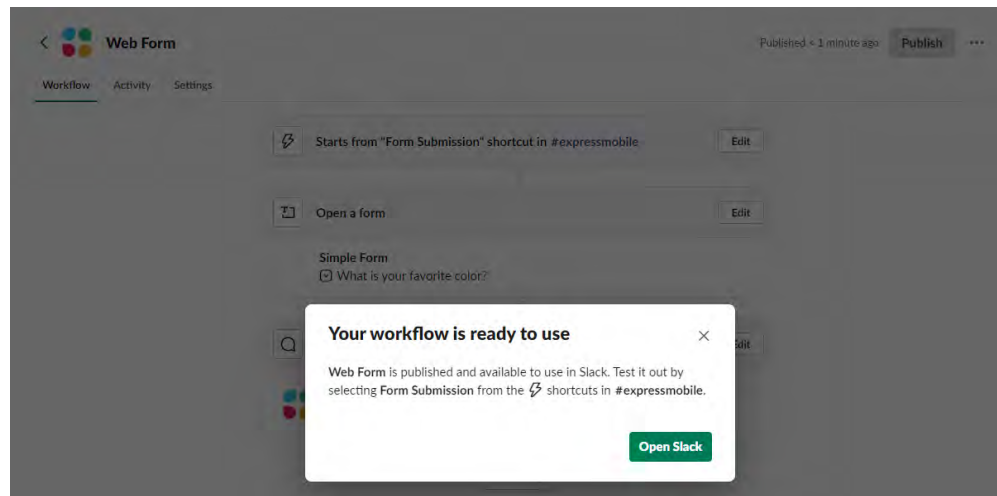
☒ Send submitted responses to a channel or to someone in a DM

#expressmobile

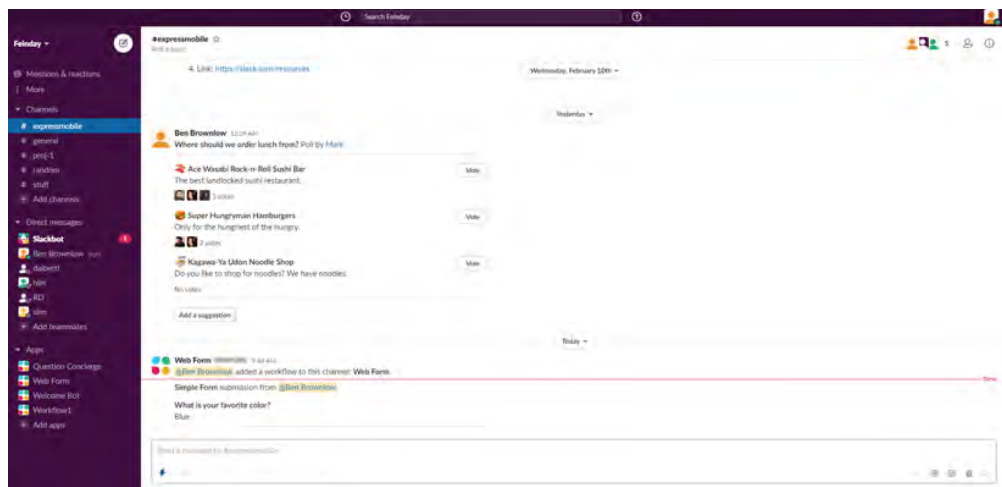
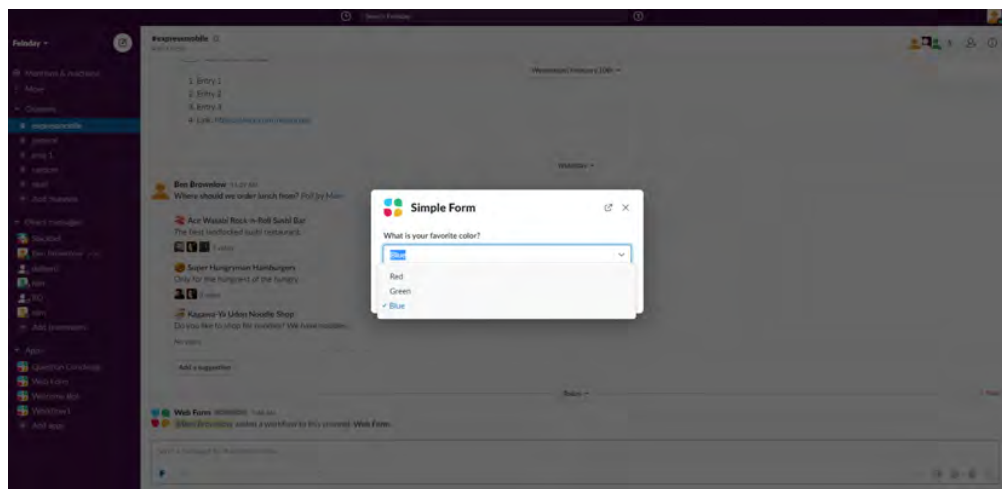
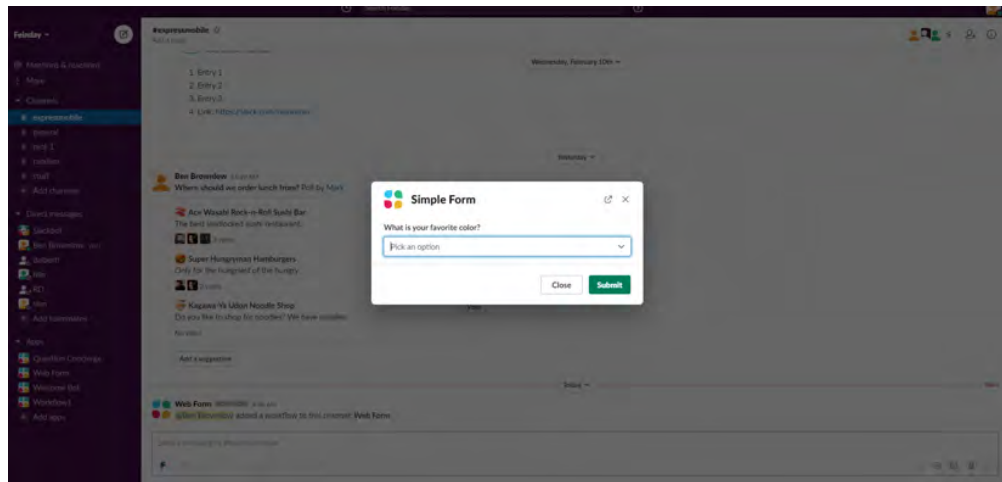
Cancel Save

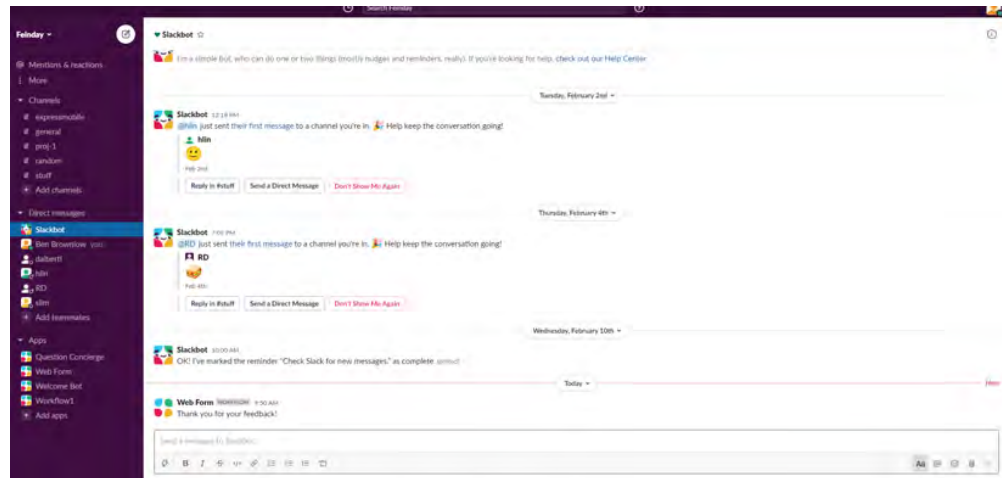
88. When a user of the Accused Instrumentalities uses the authoring tool to create a Workflow, the authoring tool accesses the computer memory to select the associated UI objects to allow the user to do so. Once selected, Slack associates the symbolic name, such as a parameter

name, with the selected defined UI object. The symbolic name is only available to corresponding UI objects. The Accused Instrumentalities store information representative of the defined UI object and related settings in a database and retrieve the information for purposes of displaying the UI object on the live web site. For example, settings related to the UI object would include the selected options for the Form Submission.









89. The Accused Instrumentalities include an authoring tool configured to build an application consisting of one or more web page views from at least a portion of the database utilizing at least one player. The player utilizes information stored in the database to generate the display of at least a portion of the one or more web pages. The Accused Instrumentalities also include an application that allows users to use Slack on mobile devices.

## Minimum requirements for using Slack

If you're curious about the technical requirements needed to run Slack, read on to learn which operating systems (OS) and browsers we support.

### Desktop apps

Operating System	Requirements
<a href="#">MacOS</a>	OS X 10.10 or above
<a href="#">Windows</a>	Windows 7 or above
<a href="#">Linux</a>	Fedora 28 or above Ubuntu LTS releases 16.04 or above Red Hat Enterprise Linux 7.0 or above

## Mobile apps

Operating System	Requirements
<a href="#">iOS</a>	iOS 12.2 or above
<a href="#">Android</a>	Android 6.0 or above

## Web browsers

Browser	Requirements
Chrome	Version 66 or above
Firefox	Version 60 or above
Safari	Version 11.0 or above
Microsoft Edge	Version 41 or above



**Note:** Google Chrome is the only browser that supports [Slack calls](#).

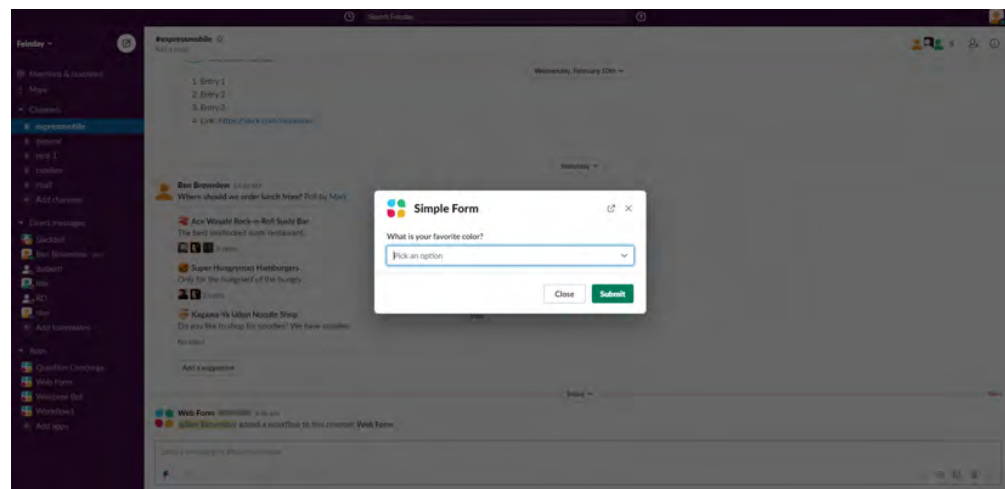
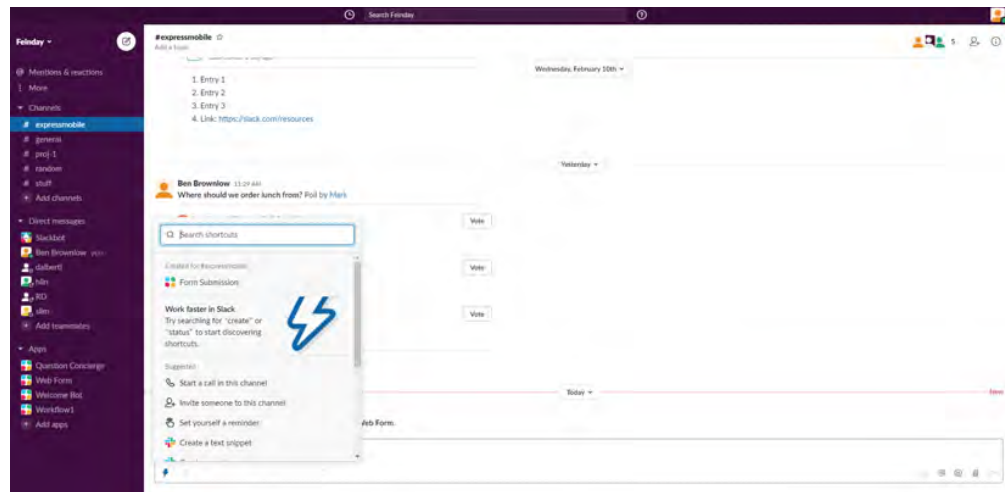
## Unsupported browsers and operating systems

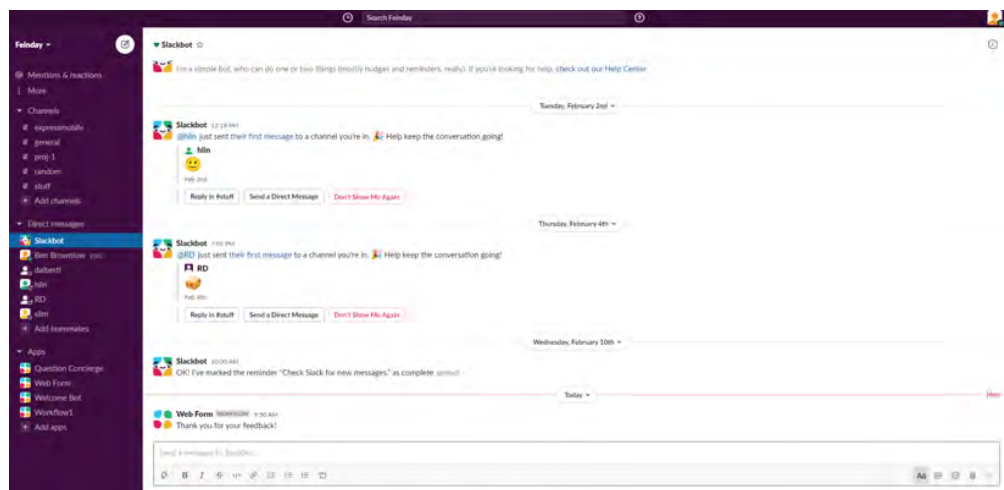
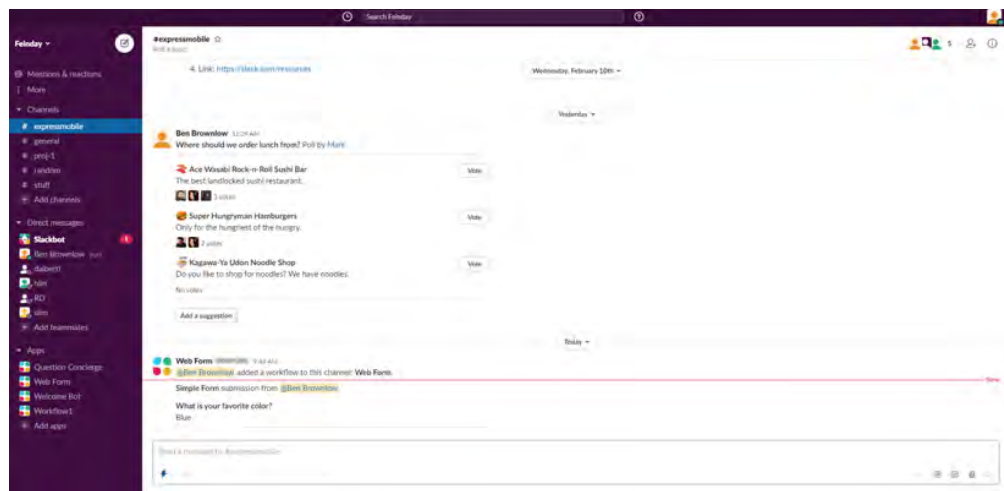
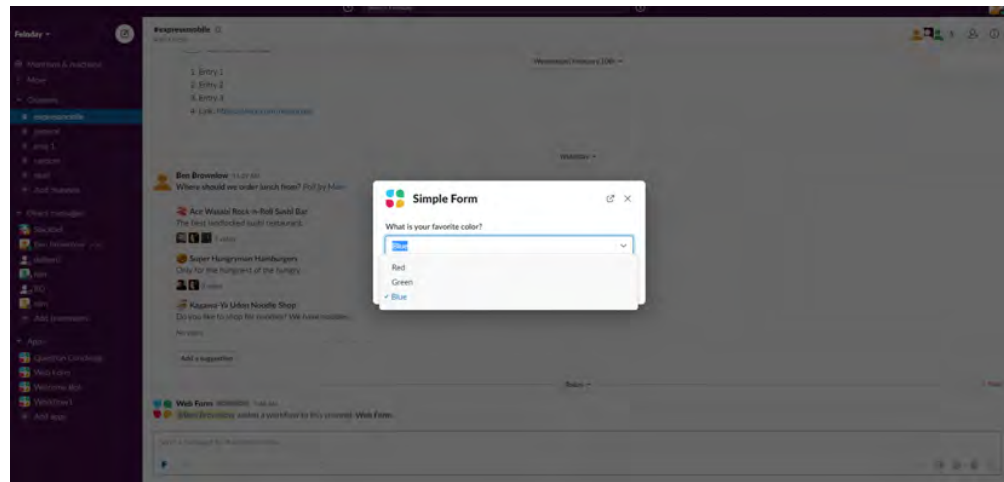
To deliver the best possible Slack experience, we keep our list of supported browsers and operating systems short. We don't fix bugs or issues for unsupported browsers, so we do our best to block them from accessing Slack.

Source: <https://slack.com/help/articles/115002037526-Minimum-requirements-for-using-Slack>

90. The application is provided, for example, in the form of software files and associated data for the web page view(s) that are stored in the Slack database. When a browser is used to access Slack, it uses a player which interacts with the application and data stored on the Slack server. The player is, for example, software provided to the device in connection with the application. The player accesses and renders the data to generate the web page viewed by the user. The player operates with the virtual machine (for example, Microsoft Internet Explorer uses the Trident or Blink virtual machine, depending on the software version) and the information stored

in the database in order to generate and display at least a portion of one or more web pages. The player includes code that is device-dependent in order to allow the environment to work across a variety of devices such as personal computers (including laptops and desktops), tablets, and mobile phones. The Accused Instrumentalities include a player or code that utilizes information stored in its database to generate for display one or more web pages as shown below.





91. The Accused Instrumentalities provide the application and the player code to the client devices to be executed on client devices. As described above and shown below, when a browser accesses Slack or views a channel on Slack, the application is provided to the device, for



example, in the form of software files and other assets. The player code operates with the virtual machine to interpret this software code and execute it locally.

## Minimum requirements for using Slack

If you're curious about the technical requirements needed to run Slack, read on to learn which operating systems (OS) and browsers we support.

### Desktop apps

Operating System	Requirements
<a href="#">MacOS</a>	OS X 10.10 or above
<a href="#">Windows</a>	Windows 7 or above
<a href="#">Linux</a>	Fedora 28 or above Ubuntu LTS releases 16.04 or above Red Hat Enterprise Linux 7.0 or above

### Mobile apps

Operating System	Requirements
<a href="#">iOS</a>	iOS 12.2 or above
<a href="#">Android</a>	Android 6.0 or above

### Web browsers

Browser	Requirements
Chrome	Version 66 or above
Firefox	Version 60 or above
Safari	Version 11.0 or above
Microsoft Edge	Version 41 or above



**Note:** Google Chrome is the only browser that supports [Slack calls](#).

### Unsupported browsers and operating systems

To deliver the best possible Slack experience, we keep our list of supported browsers and operating systems short. We don't fix bugs or issues for unsupported browsers, so we do our best to block them from accessing Slack.

Source: <https://slack.com/help/articles/115002037526-Minimum-requirements-for-using-Slack>



92. The Accused Instrumentalities allow a user to provide an input value and the corresponding input symbolic name to the web service through their device. The web service utilizes the user provided input value and the corresponding input symbolic name for generating one or more output values associated in the source code with an output symbolic name. The player receives the output symbolic name and corresponding one or more output values and provides instructions for a display of the device to present an output value in the defined UI object. The player code on the device operates with the virtual machine to execute software instructions in order to receive the output symbolic name and output value. The instructions also provide for the display of this output value in the defined UI object in order to display the appropriate data to the user. This output value is presented in the defined UI object for display on the device to the user.

`actions.value` Set by your app when you composed the blocks, this is the `value` that was specified in the interactive component when an interaction happened. For example, a `select` menu will have multiple possible values depending on what the user picks from the menu, and `value` will identify the chosen option. See the reference guide for the interactive element you're using for more info on the `value` field.

Source: <https://api.slack.com/reference/interaction-payloads/block-actions>

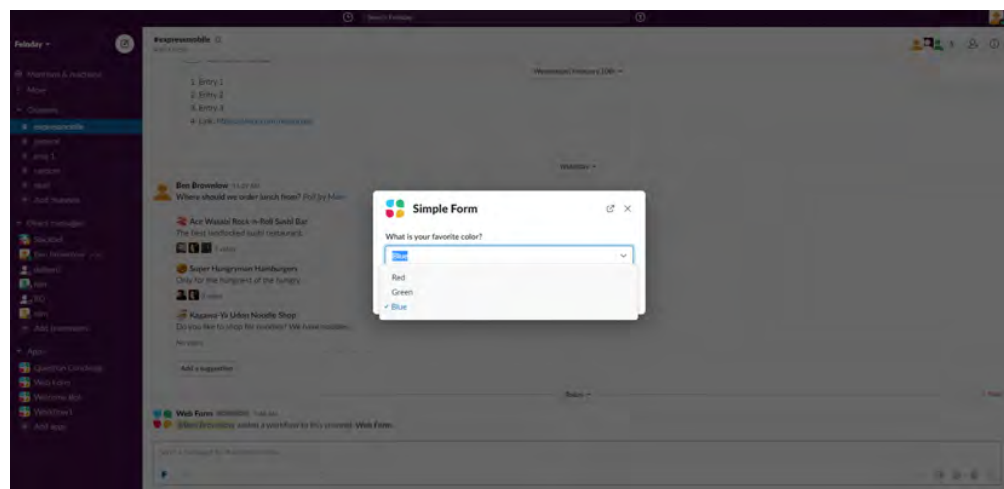
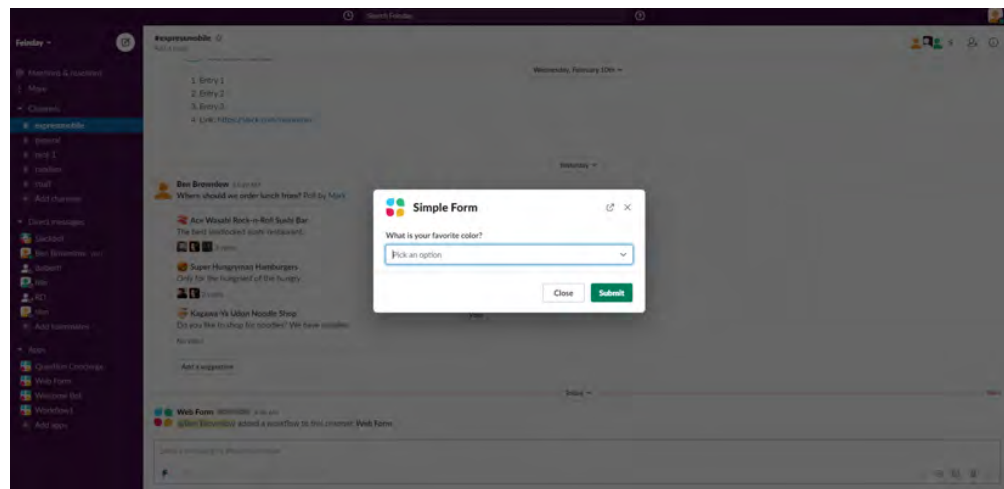
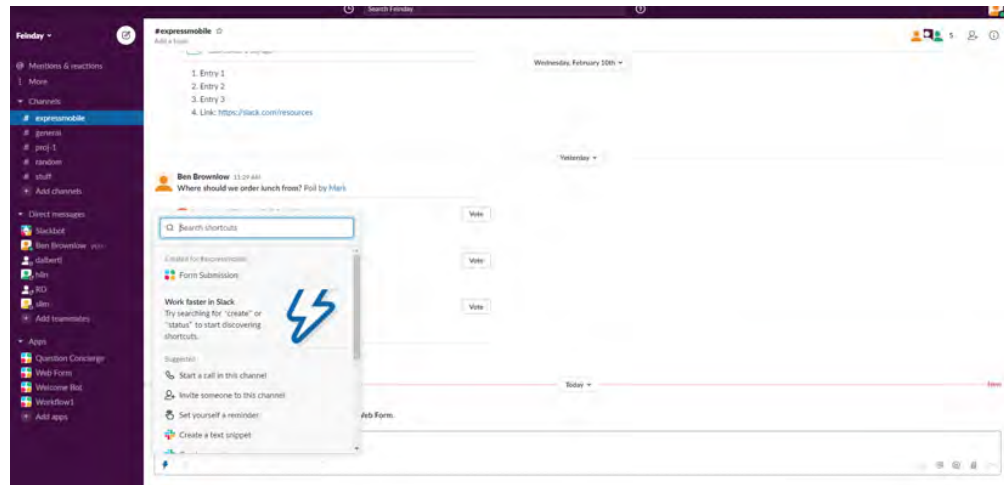
### Handling `block_actions` payloads

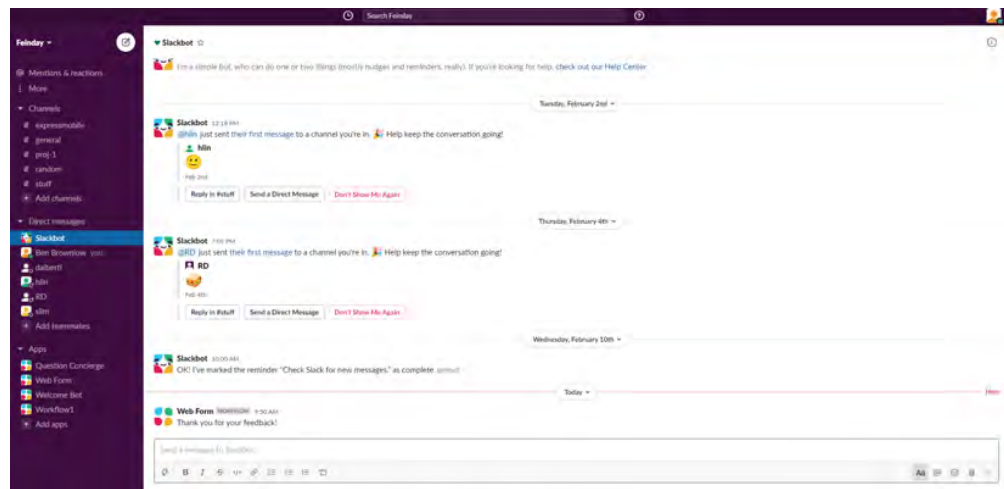
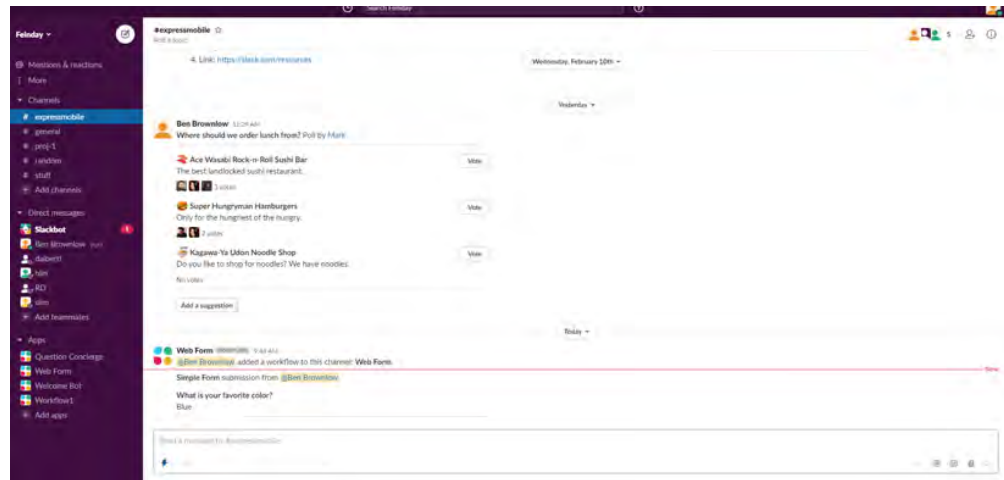
When someone uses an interactive component in your app's modal views, the app will receive a `block_actions` payload.

This **does not** apply to components included in an `input` block (see below for details about those).

Once processed, the information in the `block_actions` payload can be used to respond to the interaction, which we'll explain later.

Source: [https://api.slack.com/surfaces/modals/using#response\\_actions](https://api.slack.com/surfaces/modals/using#response_actions)





## View activity for your workflows



- 1 From your desktop, click your workspace name in the top left.
- 2 Select **Tools** from the menu, then click **Workflow Builder**.
- 3 From the **Your Workflows** tab, click a workflow to open it.
- 4 Click the **Activity** tab.
- 5 Browse the list of activity, or use the filters at the top to narrow results by status, date, or people.
- 6 Click **Show details** to see who triggered the workflow, and each step that was completed.



**Note:** You can only view workflow activity from the last 90 days.

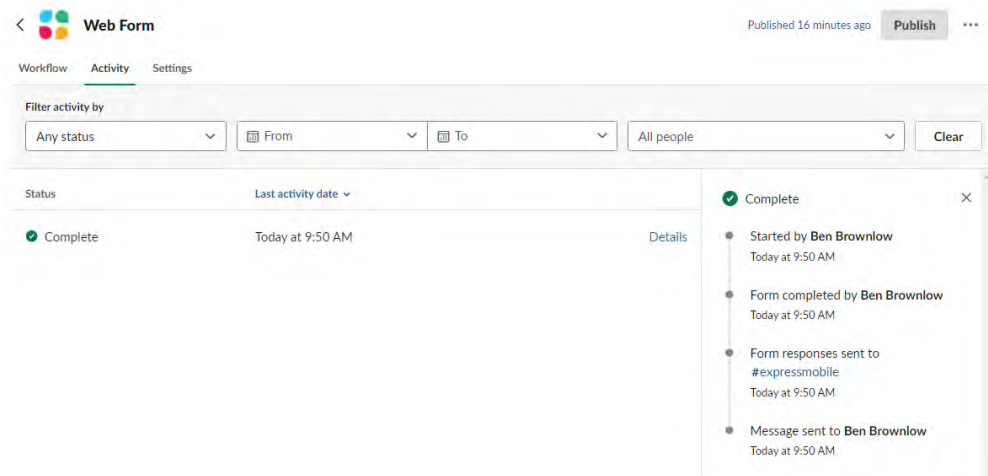
## View workflow activity in Workflow Builder

### Who can use this feature?

-  **Members** who [collaborate on workflows](#)
-  **Standard, Plus, and Enterprise Grid** plans

From the **Activity** tab in Workflow Builder, collaborators can see how people interact with their [workflows in Slack](#). View completed workflows, those that are in progress, and any workflow errors that may have occurred.

Source: <https://slack.com/help/articles/360055655493-View-workflow-activity-in-Workflow-Builder>



93. Slack was made aware of the '044 patent and its infringement thereof at least as early as December 19, 2019 when Express Mobile provided notice of Slack's infringement of the '044 patent to David Schellhase, General Counsel of Slack. Since at least the time Slack received notice, Slack has induced others to infringe at least one claim of the '044 patent under 35 U.S.C. § 271(b) by, among other things, and with specific intent or willful blindness, actively aiding and abetting others to infringe, including but not limited to Slack's clients, customers, and end users, whose use of the Accused Instrumentalities constitutes direct infringement of at least one claim of the '044 patent. In particular, Slack's actions that aid and abet others such as customers and end users to infringe include advertising and distributing the Accused Instrumentality and providing instruction materials, training, and services regarding the Accused Instrumentality. For example, Slack provides support services for the Accused Instrumentalities, including providing instructions, guides, and online materials. *See, e.g.,* slack.engineering/; slack.com/resources/; slack.com/tutorials/; slack.com/features/; slack.com/help/; <https://slack.com/help/articles/360055655493-View-workflow-activity-in-Workflow-Builder>; <https://api.slack.com/tutorials/design-expense-block-kit>; [https://www.workplace.com/help/work/1070177463385471?helpref=hc\\_global\\_nav](https://www.workplace.com/help/work/1070177463385471?helpref=hc_global_nav); [https://www.workplace.com/help/work/300047317995067?helpref=about\\_content](https://www.workplace.com/help/work/300047317995067?helpref=about_content); and all related domains and subdomains. Slack's clients, customers, and end users exercise control and obtain

1 beneficial use of the Accused Instrumentalities. For example, as demonstrated in connection with  
2 claim 1, Slack's clients, customers, and end users exercise control over and benefit from the use  
3 of Slack's Workflow Builder by using the authoring tool to define custom UI objects for interacting  
4 with other Slack end users. Slack has engaged in such actions with specific intent to cause  
5 infringement or with willful blindness to the resulting infringement because Slack has had actual  
6 knowledge of the '044 patent and knowledge that its acts were inducing infringement of the '044  
7 patent since at least the date Slack received notice that such activities infringed the '044 patent.

8         94. Slack also contributes to infringement of the '044 patent by offering to sell, and/or  
9 selling (including licensing) in the United States the Accused Instrumentalities, which are  
10 especially made or adapted for use in an infringement of the '044 patent. The Accused  
11 Instrumentalities are a material component for use in practicing the '044 patent and are not a staple  
12 article of commerce suitable for substantial non-infringing use. For example, Slack's platform,  
13 including Slack.com, the Slack Workflow Builder, the Slack mobile application, and related  
14 software and servers constitute a material portion of the invention and has no substantial non-  
15 infringing uses. Slack's clients, customers, and end users exercise control and obtain beneficial  
16 use of the Accused Instrumentalities, as shown below. For example, as demonstrated in connection  
17 with claim 1, Slack clients, customers and end users exercise control over and benefit from the use  
18 of Slack's Workflow Builder by using the authoring tool to define custom UI objects for interacting  
19 with other Slack end users. Slack is thereby liable for contributory infringement of the asserted  
20 claims of the '044 patent under 35 U.S.C. § 271(c).

21         95. Slack's infringement has damaged and continues to damage and injure Express  
22 Mobile.

23  
24                 **COUNT III - INFRINGEMENT OF U.S. PATENT NO. 9,471,287**

25         96. Plaintiff incorporates by reference the allegations contained in paragraphs 1 to 95  
26 above.

27         97. Slack has and is making, using, offering for sale, and/or selling (including licensing)  
28 in the United States Slack's platform, including Slack.com, Slack Workflow Builder, the Slack



mobile application available for Apple iOS devices (including Apple TV, iPads, iPhones and iPod Touch) on Apple's App Store, Android devices (including phones and tablets) on Google Play, and related software and servers, that afford users access to a system for generating code to provide content on a display of a device (collectively, the "Accused Instrumentalities") which infringes, either literally or under the doctrine of equivalents, one or more claims of the '287 patent in violation of 35 U.S.C. § 271(a).

98. Upon information and belief, Slack has directly infringed at least claim 1 of the '287 patent through the Accused Instrumentalities that comprise a system for generating code to provide content on the display of a device for each of its users generate code to provide content on a display of a device

99. The Accused Instrumentalities comprise a system for generating code to provide content on the display of a device for each of its users.

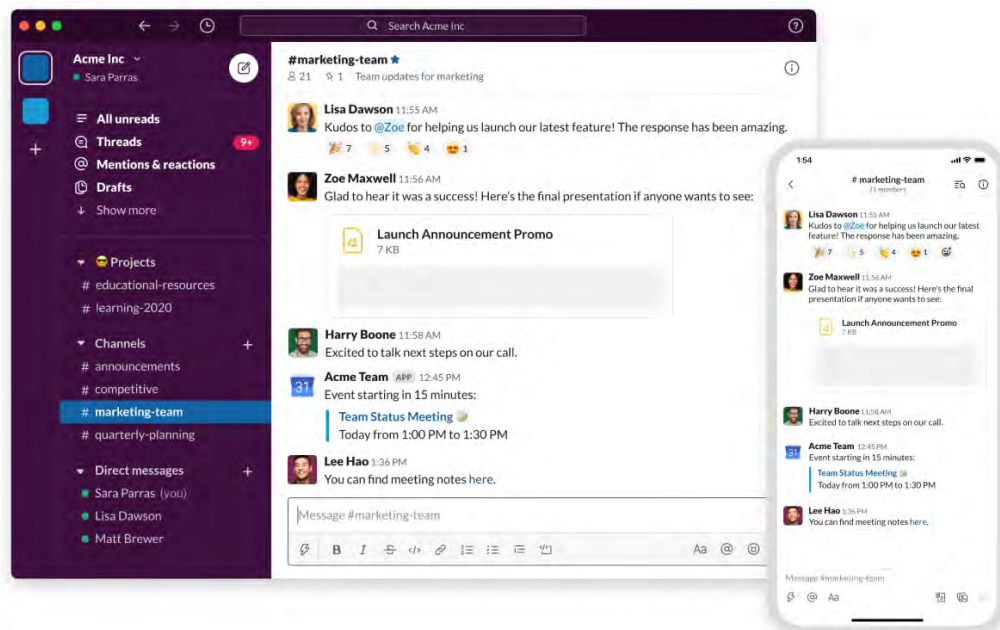
## SLACK FEATURES

# One platform for your team and your work

All the features of Slack work together so you can, too.

GET STARTED

SEE ALL FEATURES



Source: <https://slack.com/features>

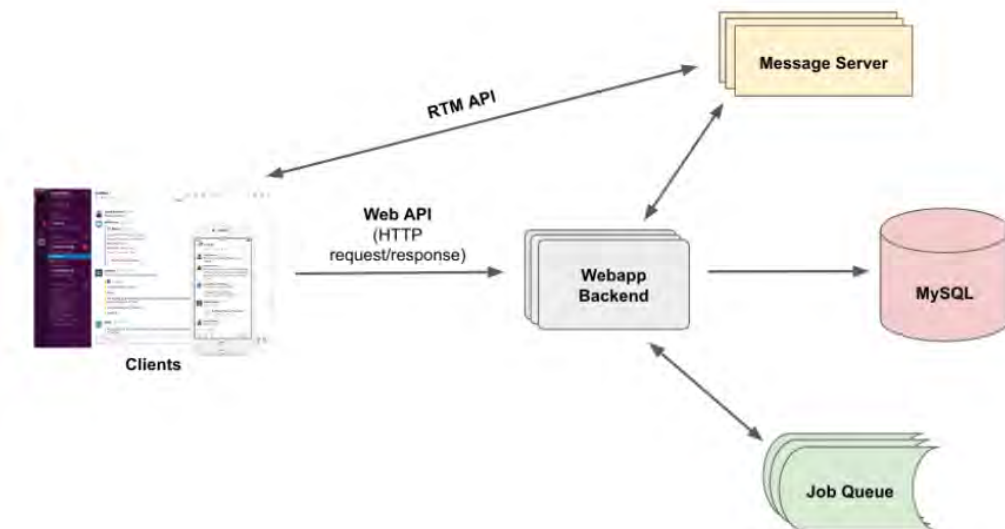
100. The Accused Instrumentalities include computer memory to store a registry of user selections and other information relating to a user's web site, web pages, and the web components or elements that are intended to be displayed.

Before we dive into shared channels, let's take a look at how Slack works at a high level. Slack implements a client-server architecture where clients (mobile, desktop, web and apps) talk to two backend systems:

- ❶ *webapp servers*, which handle HTTP request/response cycles and communicate with other systems like the main databases and job queues
- ❷ *real-time message servers*, which fan out messages, profile changes, user presence updates, and a host of other events to clients

For example, a new message posted in a channel is sent via an API call to the webapp servers, where it is sent to the message servers and is persisted in the database. The real-time message servers receive these new messages and send them to connected clients over web sockets, based on who is in the channel.

Source: <https://slack.engineering/how-slack-built-shared-channels/#:~:text=%20Slack%20implements%20a%20client-server%20architecture%20where%20clients,a%20host%20of%20other%20events%20to...%20More%20>



*Basic Slack architecture*

Source: <https://slack.engineering/how-slack-built-shared-channels/#:~:text=%20Slack%20implements%20a%20client-server%20architecture%20where%20clients,a%20host%20of%20other%20events%20to...%20More%20>


[server%20architecture%20where%20clients,a%20host%20of%20other%20events%20to...%20More%20](#)


101. The Accused Instrumentalities include symbolic names required for evoking one or more web components each related to a set of inputs and outputs of a web service obtainable over a network. The symbolic names, such as parameter names, are character strings that do not contain either a persistent address or pointer to an output value accessible to the web service.

102. The Accused Instrumentalities include a feature called Workflow Builder, which offers a set of tools to help users automate processes into workflows (or custom applications) in their Slack workspace. Workflows require the use of a symbolic name, or parameter value, to evoke a web component related to a set of inputs and outputs of a web service obtainable over a network.

## Guide to Workflow Builder

### Who can use this feature?

 By default, all **members** of a workspace can [create workflows](#). Any **member** or **guest** can use workflows added to channels they belong to.

 **Standard, Plus, and Enterprise Grid** plans

[Workflow Builder](#) offers a set of tools — no coding required! — to help you automate routine processes into workflows in your Slack workspace. Workflows in Slack can be as simple or as complex as you'd like, and can even be connected to other apps and services you use to get work done.

Read on to learn about Workflow Builder, then use our tutorials as a guide to build your workflow.

Source: <https://slack.com/help/articles/360035692513-Guide-to-Workflow-Builder>

## Modals

Modals are dedicated windows that allow your app to collect input from users and display dynamic information in a focused space. Modals must be initiated by a user (with features like buttons, slash commands, and other invocative actions).

Source: <https://api.slack.com/tutorials/design-expense-block-kit>  
<https://api.slack.com/tutorials/design-expense-block-kit>

## Workflow Builder glossary

<b>Trigger</b>	How your workflow will start. You can choose a ⚡ <b>shortcuts menu</b> trigger to launch your workflow from the menu in a channel, or another trigger to start your workflow automatically when a certain action is taken.
<b>Step</b>	How people interact with your workflow. You can add steps that send messages or custom forms, or use <a href="#">steps from Slack apps</a> to link your workflow to other tools.
<b>Variable</b>	Information submitted to your workflow that can be referenced in steps. More variables will be available to choose from as you add steps to your workflow.
<b>Collaborator</b>	A member of your workspace with permission to manage your workflow. Any collaborator can change or remove steps, manage other collaborators, and unpublish or delete a workflow.






Source: <https://slack.com/help/articles/360035692513-Guide-to-Workflow-Builder>

## Set up your workflow

While you'll first be presented with picking a name for your workflow, the more important decision you'll need to make is how your workflow will start.

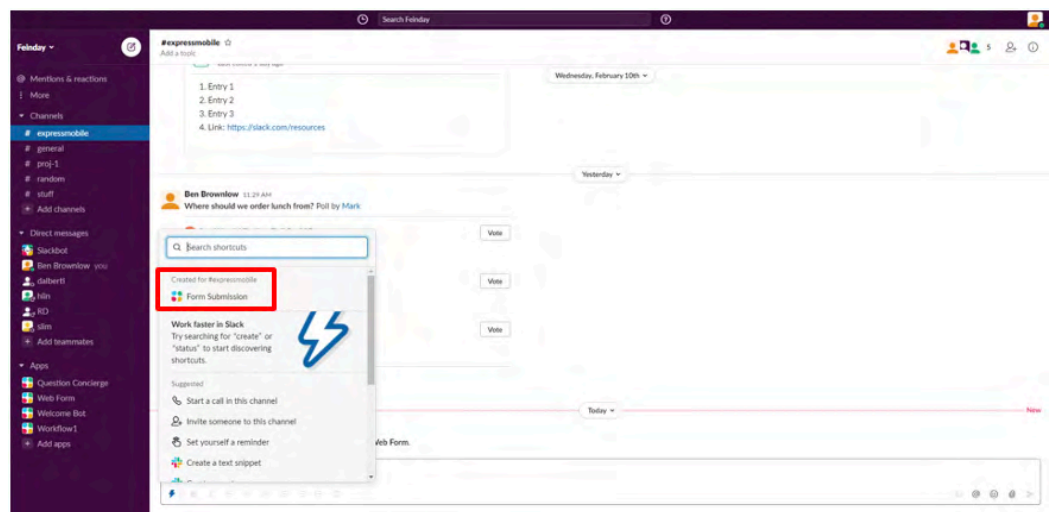
### Choose a trigger

The trigger you pick for your workflow determines how it will start. Workflows are channel-specific, and can only be used in the channel you choose.

Trigger	How it works
 Shortcuts menu	Your workflow starts when someone selects it from the shortcuts menu.
 New channel member	Your workflow starts automatically when someone joins the channel.
 Emoji reaction	Your workflow starts automatically when someone in the channel adds a certain <a href="#">emoji reaction</a> to a message.
 Scheduled date & time	Your workflow starts automatically at a set date and time.
 Webhook*	Your workflow starts automatically <a href="#">when a custom webhook sends a web request to Slack</a> from another app or service.

\*Creating workflows with webhooks requires some technical expertise.

Source: <https://slack.com/help/articles/360053571454-Set-up-a-workflow-in-Slack>





103. The Accused Instrumentalities include an address of the web service.

## Network settings

---

Slack must have a persistent connection between our messaging server and members' apps or browsers. To do so, Slack uses WebSockets over port 443.

Proxies and firewalls can sometimes interrupt this connection. If someone using Slack can't connect from a specific location, or if there's a WebSocket failure in our [Slack connection test](#), you'll need to adjust your proxy or firewall to keep the connection to Slack open. Here's how:

- ✔ Visit <https://my.slack.com/help/urls> and add all specified URLs to your allowlist. If your environment requires access to multiple workspaces and/or orgs, visit that page from each workspace and/or org.
- ✔ Check if your proxy is running SSL decryption. If it is, the proxy must either support WebSockets, or you'll need to exempt the following domains: \*.lb.slack-msgs.com wss-primary.slack.com wss-backup.slack.com wss-mobile.slack.com

Network administrators can restrict access to specific Slack workspaces using [proxy header injection](#) on their network. The header injection solution has the same network requirements as above.



```

1      "*.chime.aws",
2      "*.lb.slack-msgs.com",
3      "a.slack-edge.com",
4      "a.slack-imgs.com",
5      "admin.slack.com",
6      "alpha.slack.com",
7      "api.slack.com",
8      "app.slack.com",
9      "assets.slack.com",
10     "auth.slack.com",
11     "avatars.slack-edge.com",
12     "b.slack-edge.com",
13     "b.slack-imgs.com",
14     "beta.slack.com",
15     "blog.slack.com",
16     "ca.slack-edge.com",
17     "downloads.slack-edge.com",
18     "edgeapi.slack.com",
19     "email.slack.com",
20     "email2.slack.com",
21     "email3.slack.com",
22     "email4.slack.com",
23     "emoji.slack-edge.com",
24     "example.slack.com",
25     "feinday.slack.com",
26     "files-edge.slack.com",
27     "files-origin.slack.com",
28     "files.slack.com",
29     "global-upload-edge.slack.com",
30     "go-beta.slack.com",
31     "go-debug.slack.com",
32     "go.slack.com",
33     "help.slack.com",
34     "hooks.slack.com",
35     "join.slack.com",
36     "my.slack.com",
37     "oauth2.slack.com",
38     "platform-tls-client.slack.com",
39     "platform.slack-edge.com",
40     "slack-email.slack.com",
41     "slack-files.com",
42     "slack-imgs.com",
43     "slack-infra.slack.com",
44     "slack-redirect.com",
45     "slack-redirect.net",
46     "slack.com",
47     "slack.global.ssl.fastly.net",
48     "slackb.com",
49     "spellcheck.slack-edge.com",
50     "status.slack.com",
51     "try.slack.com",
52     "universal-upload-edge.slack.com",
53     "upload.slack.com",
54     "wss-backup.slack.com",
55     "wss-mobile.slack.com",
56     "wss-primary.slack.com"

```

Source: <https://slack.com/help/articles/360001603387-Manage-Slack-connection-issues>

104. The Accused Instrumentalities include an authoring tool configured to define a UI object for presentation on the display. The UI object corresponds to a web component which consists of an input of the web service and an output of the web service. For example, Workflow

Builder allows a user to define a UI object corresponding to a web component for presentation on the display

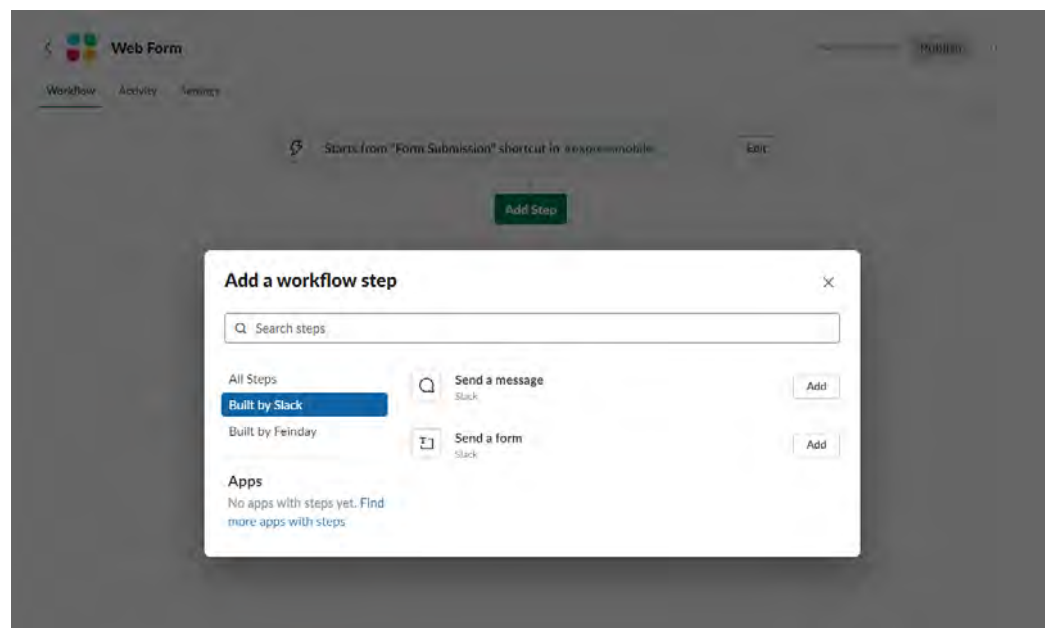
### Form steps

Add **forms** to your workflow to organize and collect information. Forms can include space for custom responses, or prompt people to choose options from a list. Once a form has been completed, you can choose to send responses to a channel or a specific person via DM.



**Note:** Form responses cannot be edited once they've been submitted via Workflow Builder.

Source: <https://slack.com/help/articles/360053571474-Add-steps-to-a-workflow>



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12

**Send a form**

Title  
Simple Form

Questions

Add a question  
What is your favorite color?

Choose a question type  
Select from a list

Short answer  
Long answer  
Select from a list  
Select a person  
Select a channel or DM

Default selection  
No selection

☒ Make this required

Add Question

☒ Send submitted responses to a channel or to someone in a DM

Cancel Save

13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28

**Send a form**

Questions

Add a question  
What is your favorite color?

Choose a question type  
Select from a list

List Items  
Red  
Green  
Blue

Add List Item

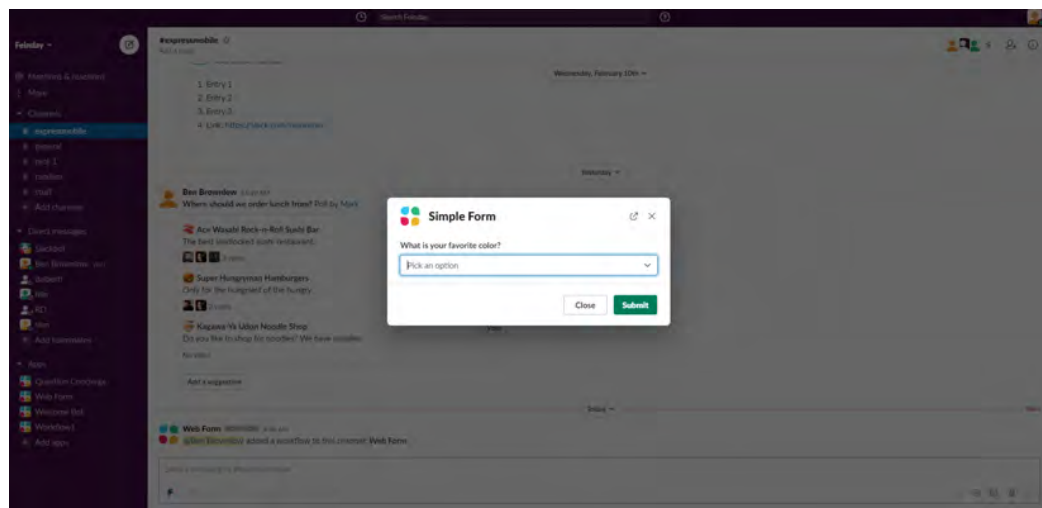
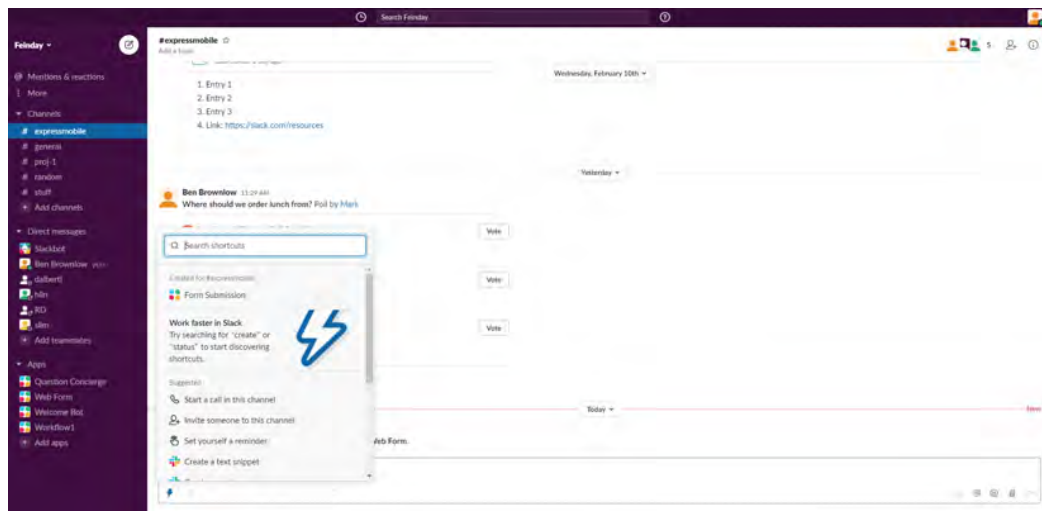
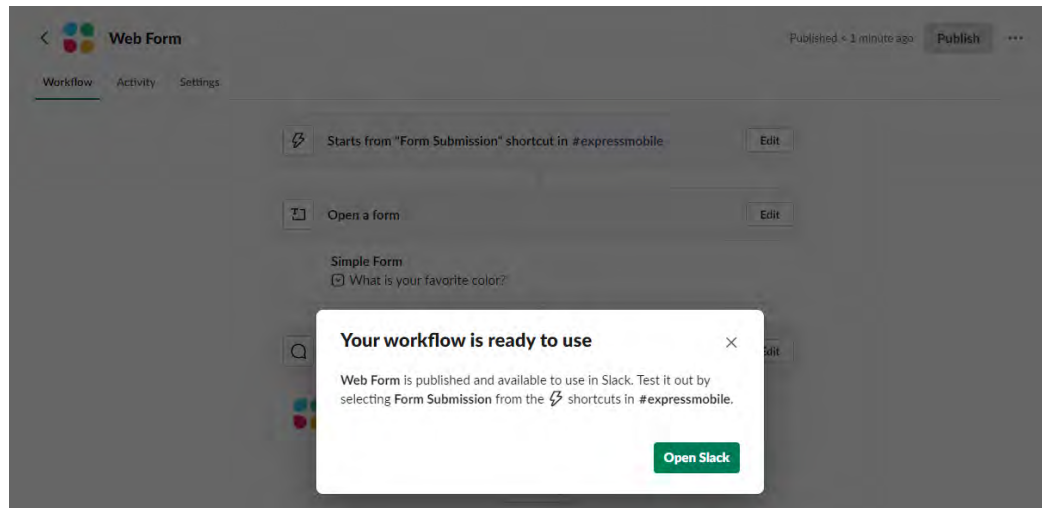
Default selection  
No selection

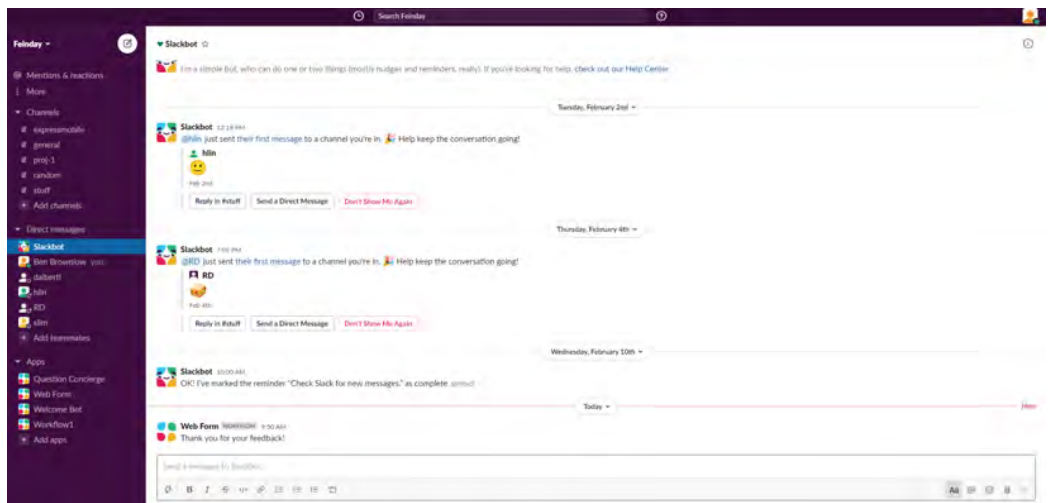
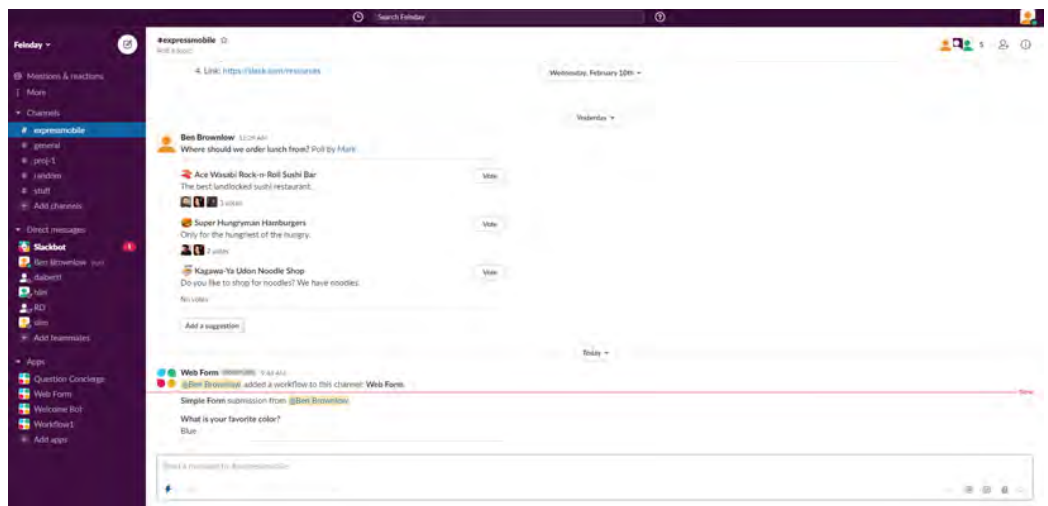
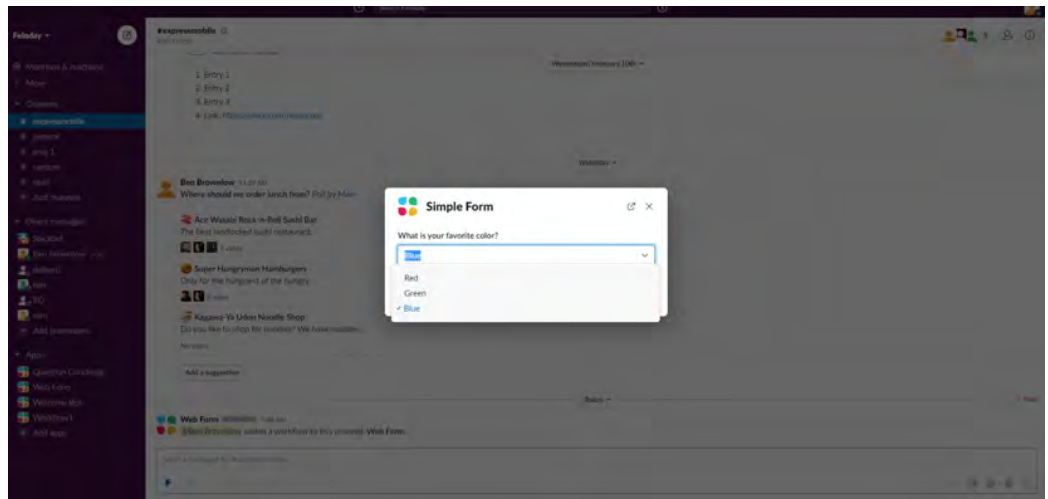
☒ Make this required

Add Question

☒ Send submitted responses to a channel or to someone in a DM  
# expressmobile

Cancel Save





105. The Accused Instrumentalities include an authoring tool configured to define a UI object for presentation on a display. Each defined UI object can be selected by a user of the authoring tool.

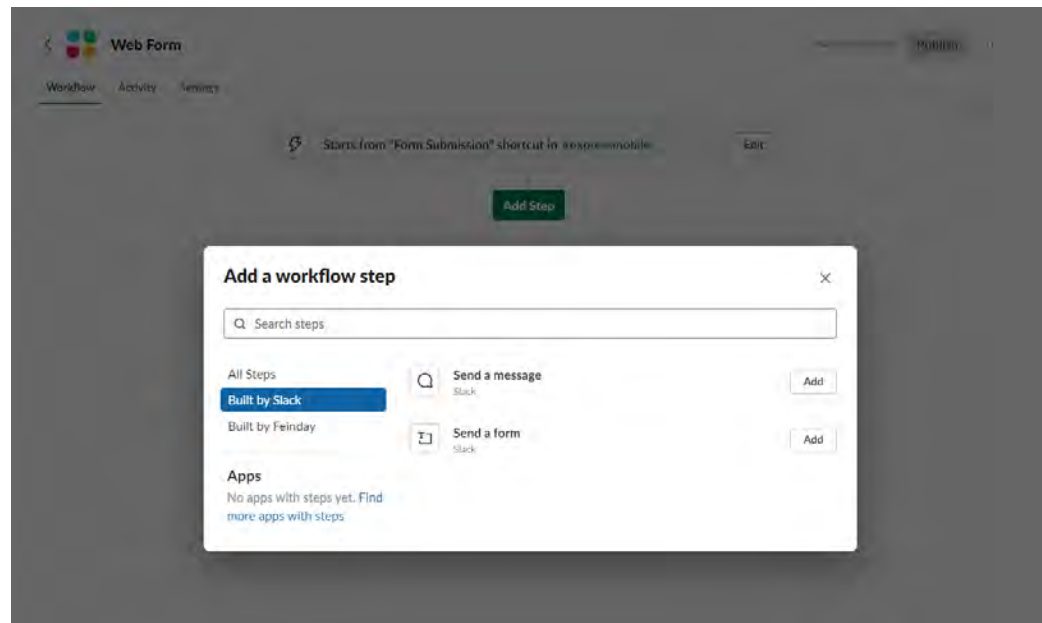
### Form steps

Add **forms** to your workflow to organize and collect information. Forms can include space for custom responses, or prompt people to choose options from a list. Once a form has been completed, you can choose to send responses to a channel or a specific person via DM.



**Note:** Form responses cannot be edited once they've been submitted via Workflow Builder.

Source: <https://slack.com/help/articles/360053571474-Add-steps-to-a-workflow>





**Web Form**

**Send a form**

Title: Simple Form

Questions

Add a question: What is your favorite color?

Choose a question type: **Select from a list**

Short answer

Long answer

✓ Select from a list

Select a person

Select a channel or DM

Default selection: No selection

☒ Make this required

Add Question

☒ Send submitted responses to a channel or to someone in a DM

Cancel Save

**Web Form**

**Send a form**

Questions

Add a question: What is your favorite color?

Choose a question type: Select from a list

List Items:

Red

Green

Blue

Add List Item

Default selection: No selection

☒ Make this required

Add Question

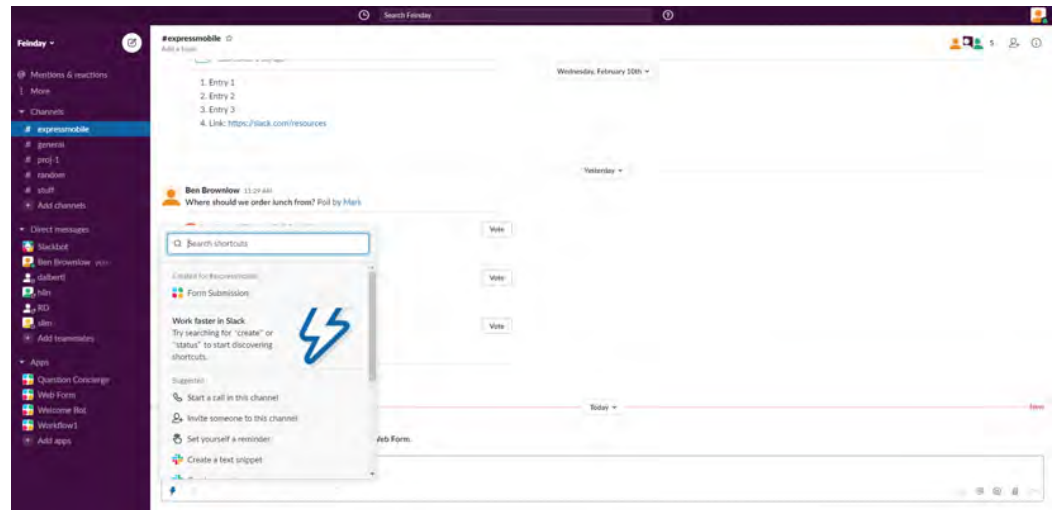
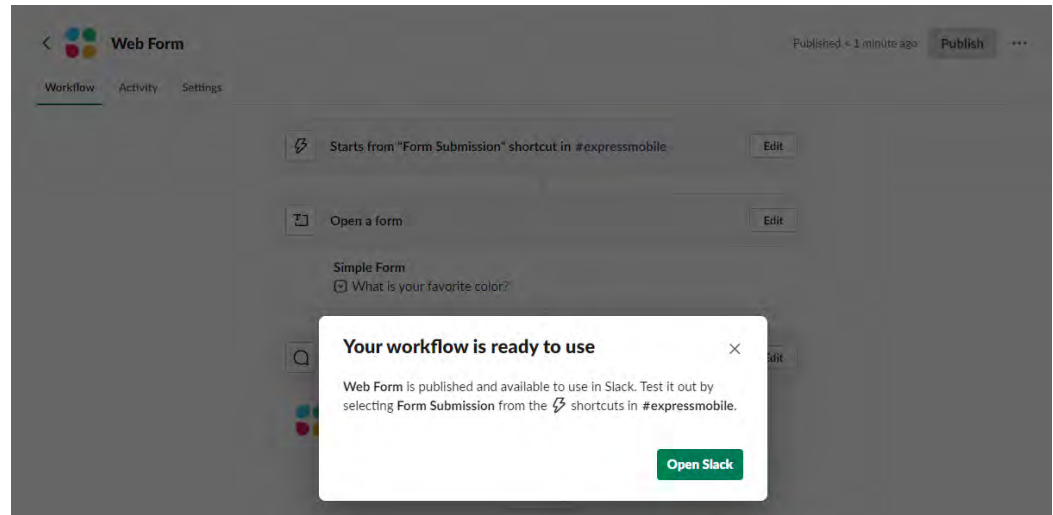
☒ Send submitted responses to a channel or to someone in a DM

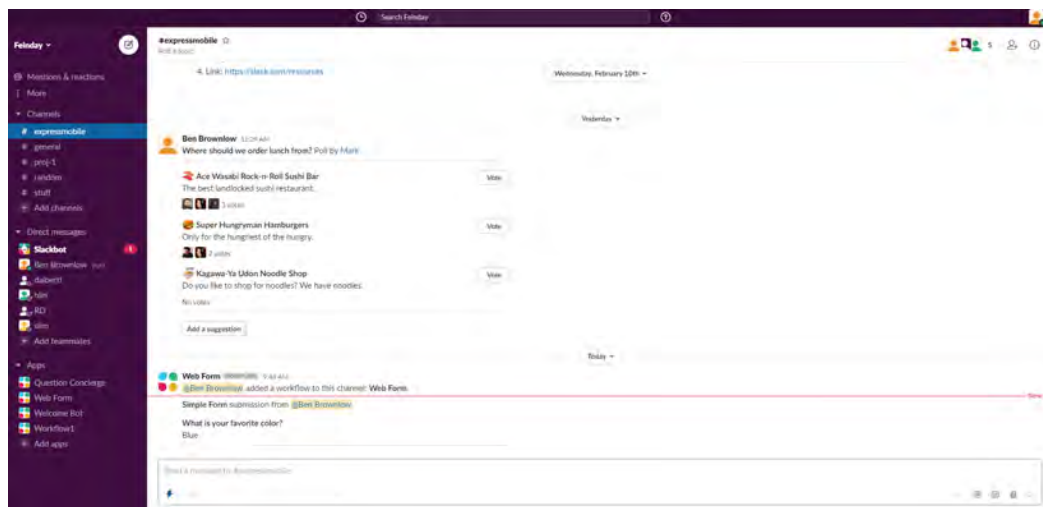
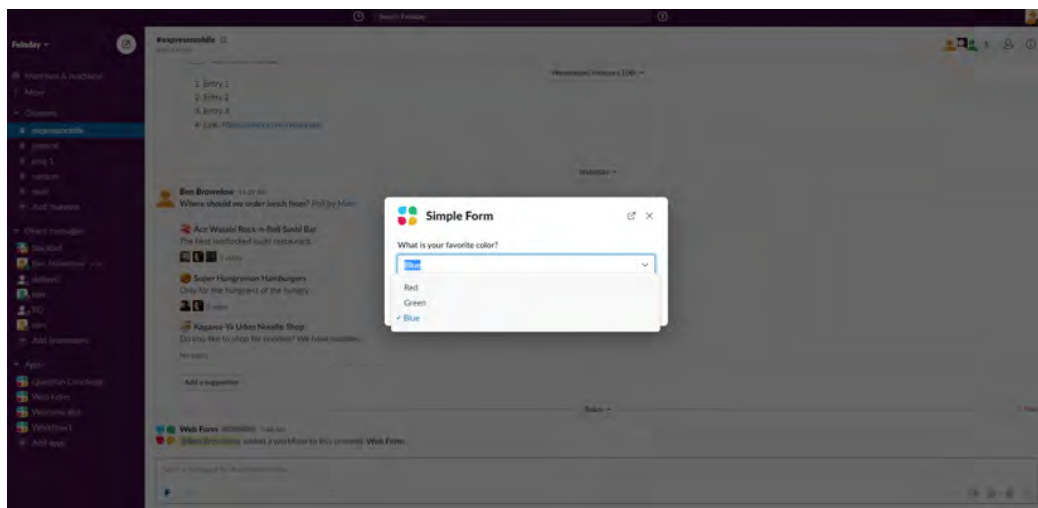
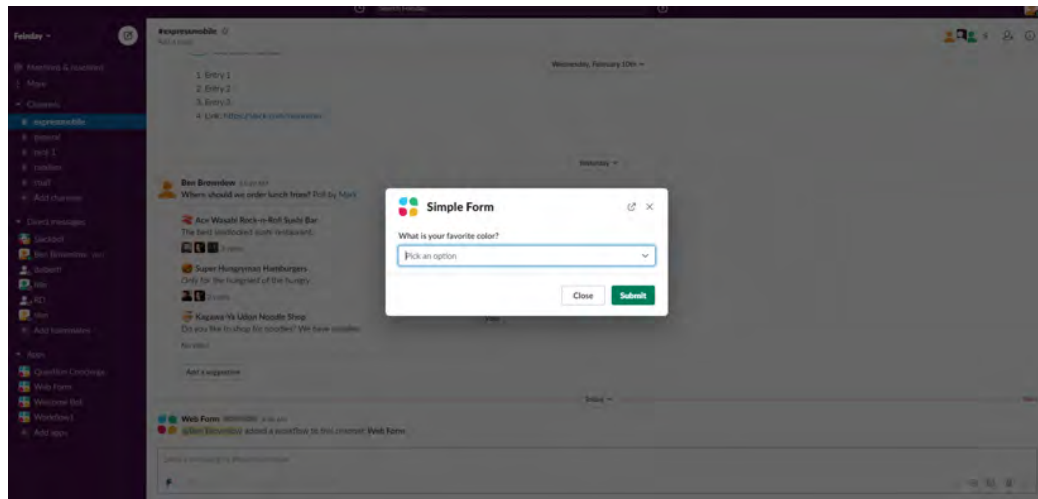
# expressmobile

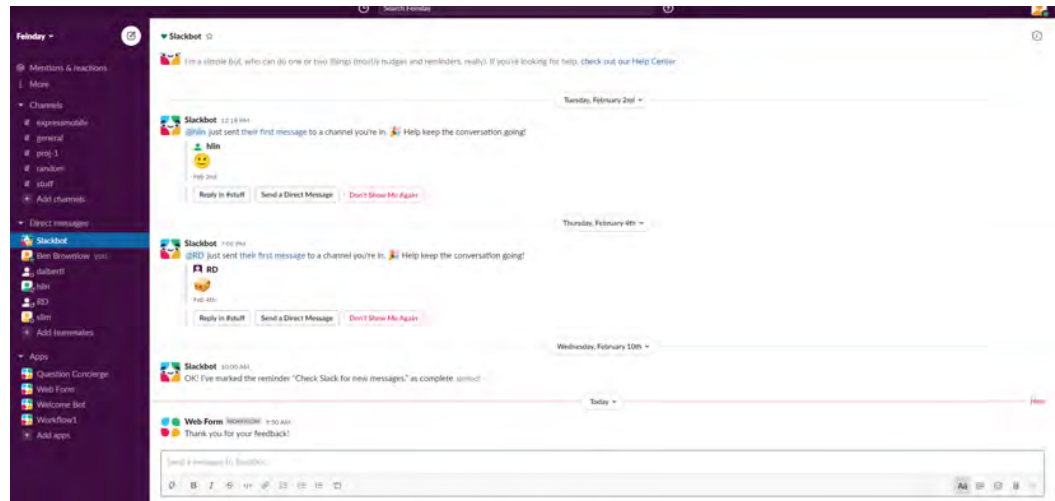
Cancel Save

106. When a user of the Accused Instrumentalities uses the authoring tool to create a Workflow, the authoring tool accesses the computer memory to select the associated UI objects

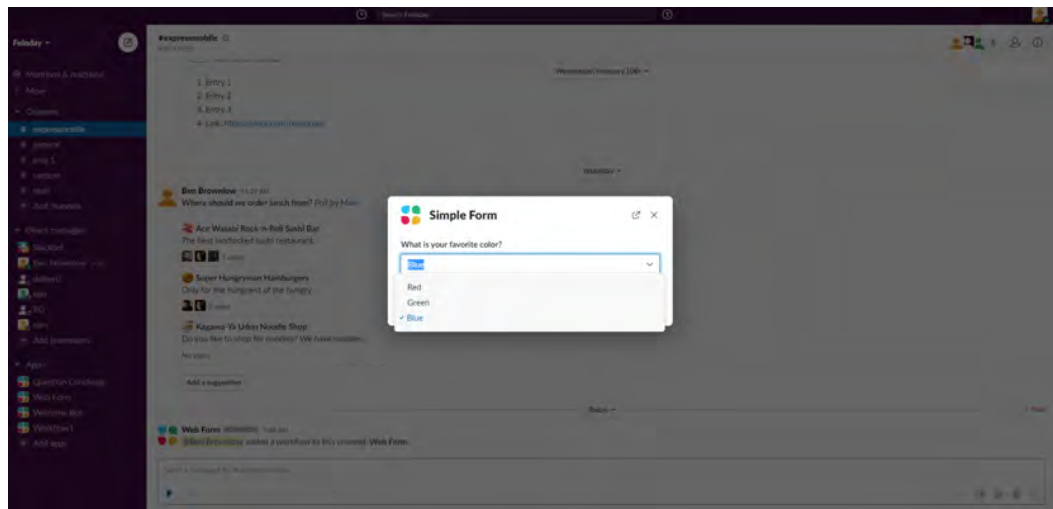
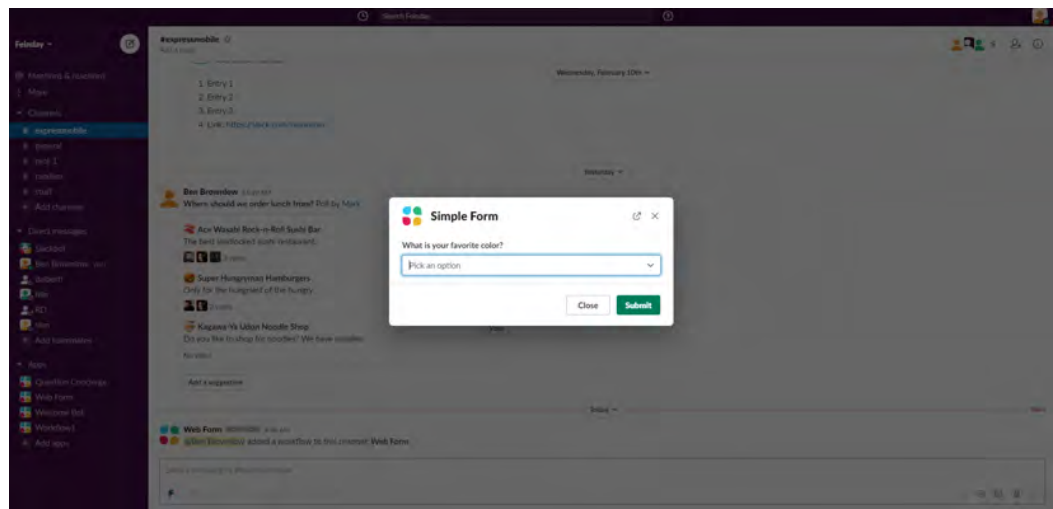
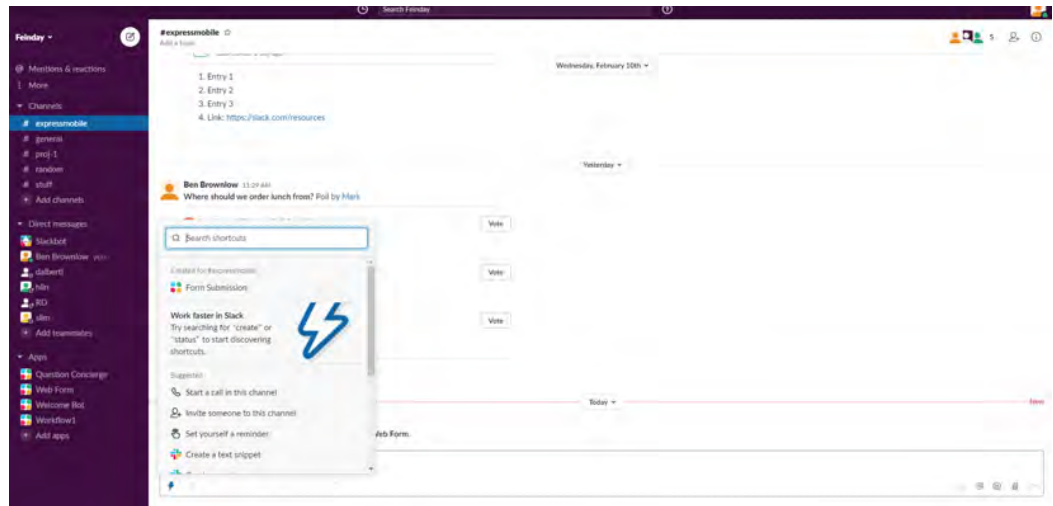
to allow the user to do so. Once selected, Slack associates the symbolic name, such as a parameter name, with the selected defined UI object. The symbolic name is only available to corresponding UI objects.

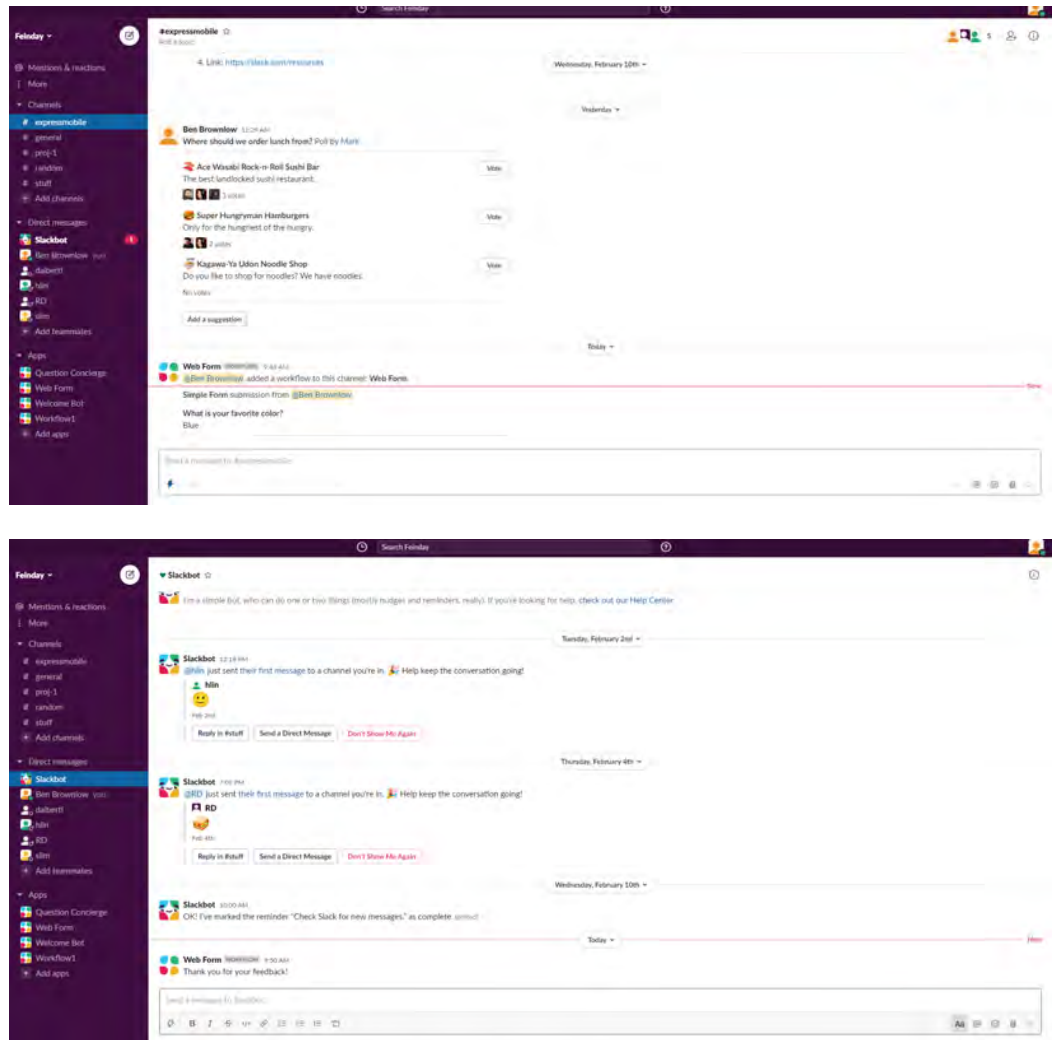






107. The application is provided, for example, in the form of software files and associated data for the web page view(s) that are stored in the Slack database. When a browser is used to access Slack, it uses a player which interacts with the application and data stored on the Slack server. The player is, for example, software provided to the device in connection with the application. The player accesses and renders the data to generate the web page viewed by the user. The player operates with the virtual machine (for example, Microsoft Internet Explorer uses the Trident or Blink virtual machine, depending on the software version) and the information stored in the database in order to generate and display at least a portion of one or more web pages. The player includes code that is device-dependent in order to allow the environment to work across a variety of devices such as personal computers (including laptops and desktops), tablets, and mobile phones. The Accused Instrumentalities include a player or code that utilizes information stored in its database to generate for display one or more web pages as shown below.





108. The Accused Instrumentalities provide the application and the player to the device to be executed on the device. When a user of the device provides one or more input values associated with an input symbolic name to an input of the defined UI object, the device provides the user provided one or more input values and corresponding input symbolic name to the web service. The web service utilizes the input symbolic name and the user provided one or more input values for generating one or more output values having an associated output symbolic name.



`actions.value`

Set by your app when you composed the blocks, this is the `value` that was specified in the interactive component when an interaction happened. For example, a `select` menu will have multiple possible values depending on what the user picks from the menu, and `value` will identify the chosen option. See the [reference guide](#) for the interactive element you're using for more info on the `value` field.

Source: <https://api.slack.com/reference/interaction-payloads/block-actions>

## Handling `block_actions` payloads

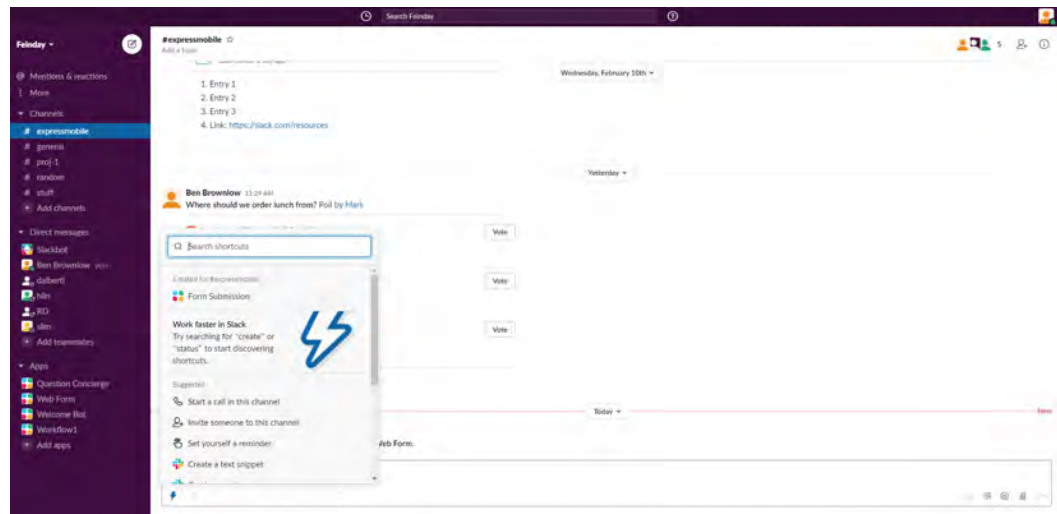
When someone uses an [interactive component](#) in your app's modal views, the app will receive a `block_actions` payload.

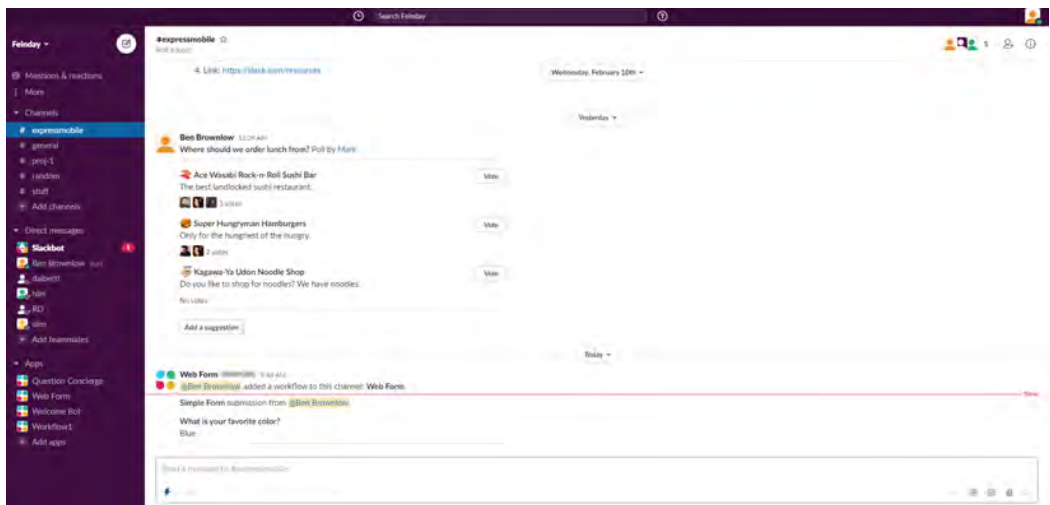
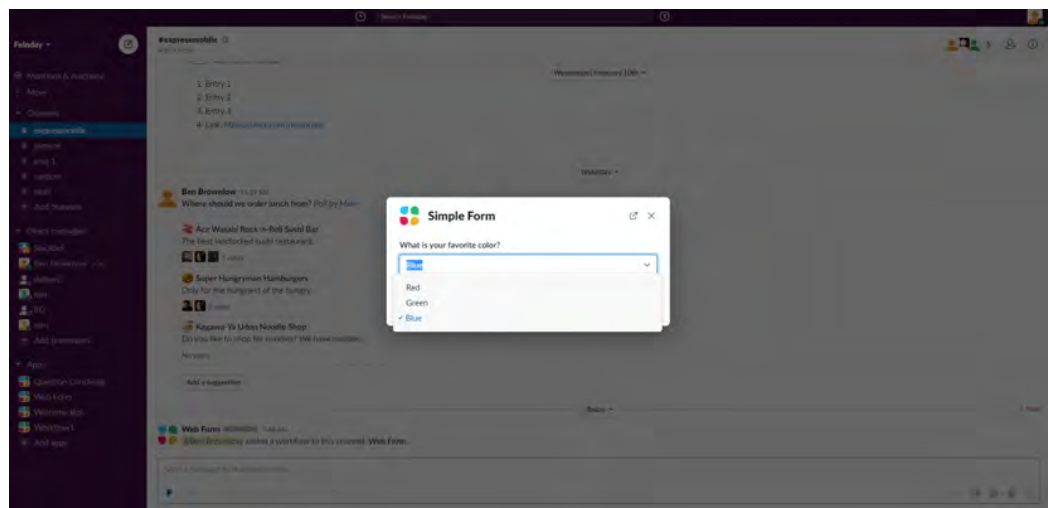
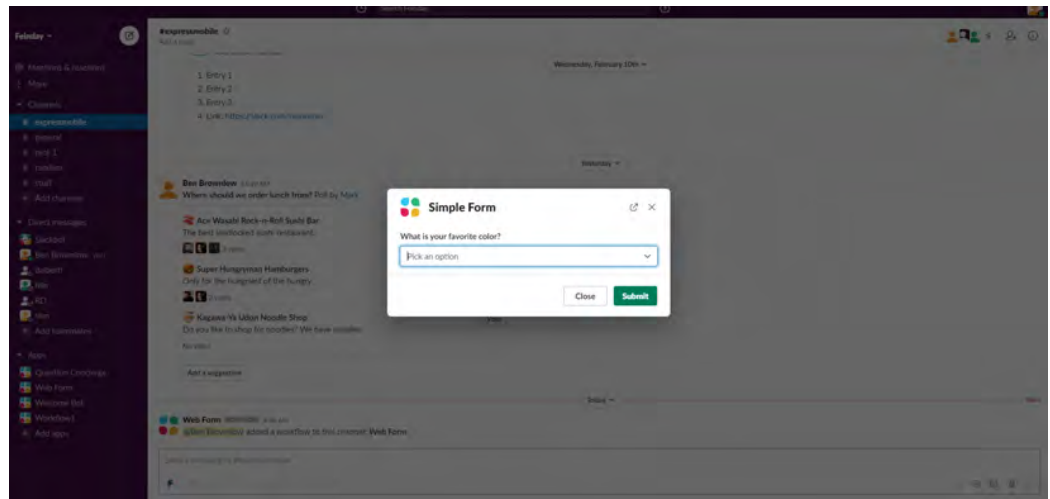
This **does not** apply to components included in an `input` block (see [below](#) for details about those).

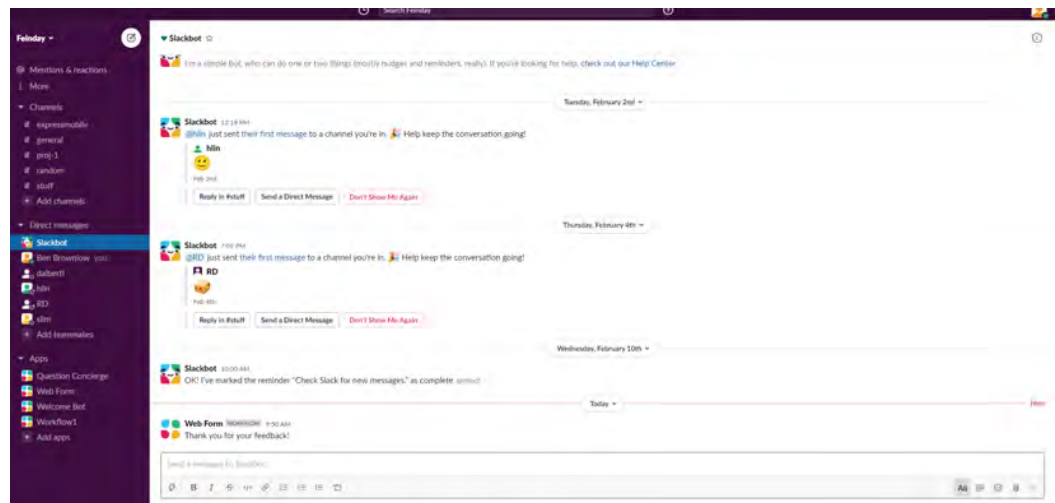
Once processed, the information in the `block_actions` payload can be used to respond to the interaction, [which we'll explain later](#).

Source: [https://api.slack.com/surfaces/modals/using#response\\_actions](https://api.slack.com/surfaces/modals/using#response_actions)

109. The player receives the output symbolic name and corresponding one or more output values and provides instructions for the display of the device to present an output value in the defined UI object.







110. The Accused Instrumentalities also include an application that allows users to use Slack on mobile devices.

## Minimum requirements for using Slack

If you're curious about the technical requirements needed to run Slack, read on to learn which operating systems (OS) and browsers we support.

### Desktop apps

Operating System	Requirements
<a href="#">MacOS</a>	OS X 10.10 or above
<a href="#">Windows</a>	Windows 7 or above
<a href="#">Linux</a>	Fedora 28 or above Ubuntu LTS releases 16.04 or above Red Hat Enterprise Linux 7.0 or above

## Mobile apps

Operating System	Requirements
<a href="#">iOS</a>	iOS 12.2 or above
<a href="#">Android</a>	Android 6.0 or above

## Web browsers

Browser	Requirements
Chrome	Version 66 or above
Firefox	Version 60 or above
Safari	Version 11.0 or above
Microsoft Edge	Version 41 or above



**Note:** Google Chrome is the only browser that supports [Slack calls](#).

## Unsupported browsers and operating systems

To deliver the best possible Slack experience, we keep our list of supported browsers and operating systems short. We don't fix bugs or issues for unsupported browsers, so we do our best to block them from accessing Slack.

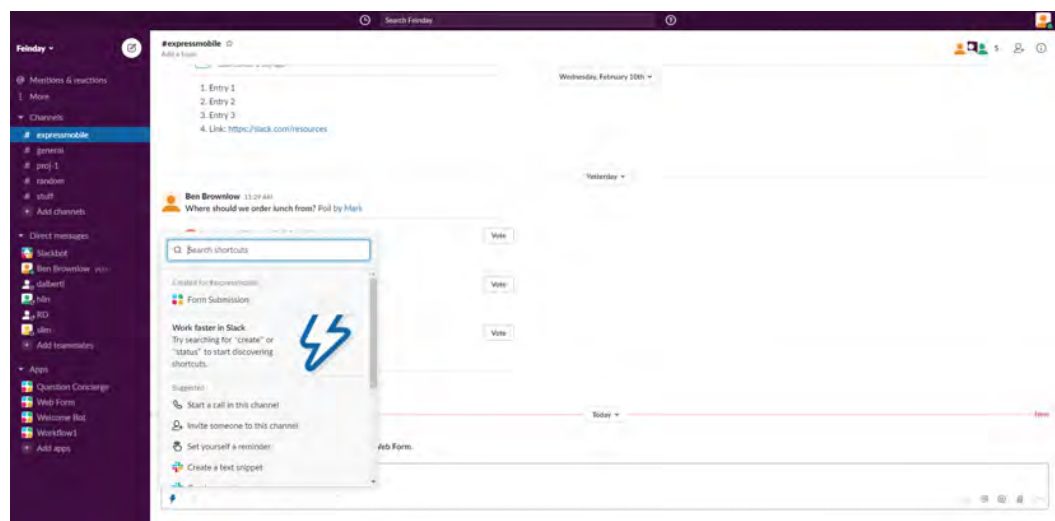
Source: <https://slack.com/help/articles/115002037526-Minimum-requirements-for-using-Slack>

111. The application is provided, for example, in the form of software files and associated data for the web page view(s) that are stored in the Slack database. When a browser is used to access Slack, it uses a player which interacts with the application and data stored on the Slack server. The player is, for example, software provided to the device in connection with the application. The player accesses and renders the data to generate the web page viewed by the user. The player operates with the virtual machine (for example, Microsoft Internet Explorer uses the Trident or Blink virtual machine, depending on the software version) and the

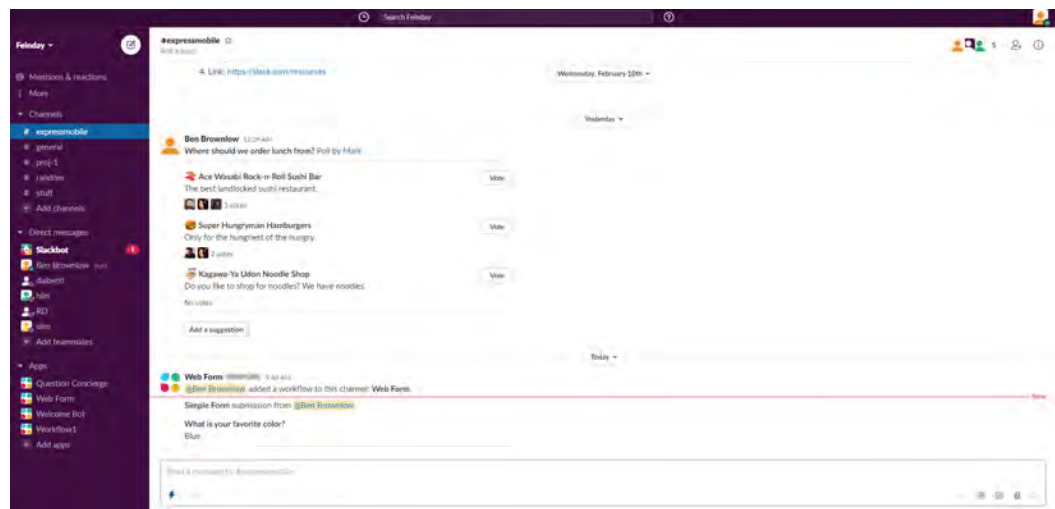
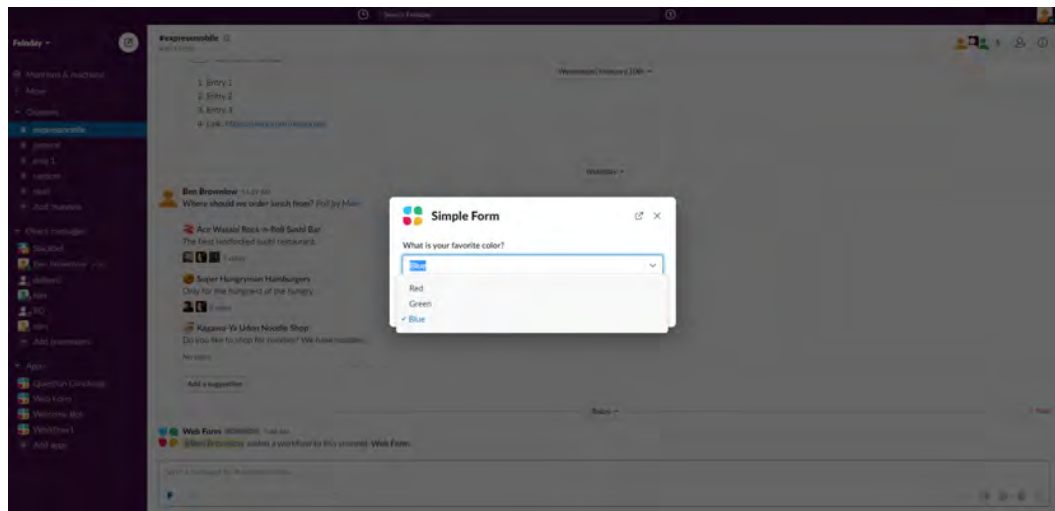
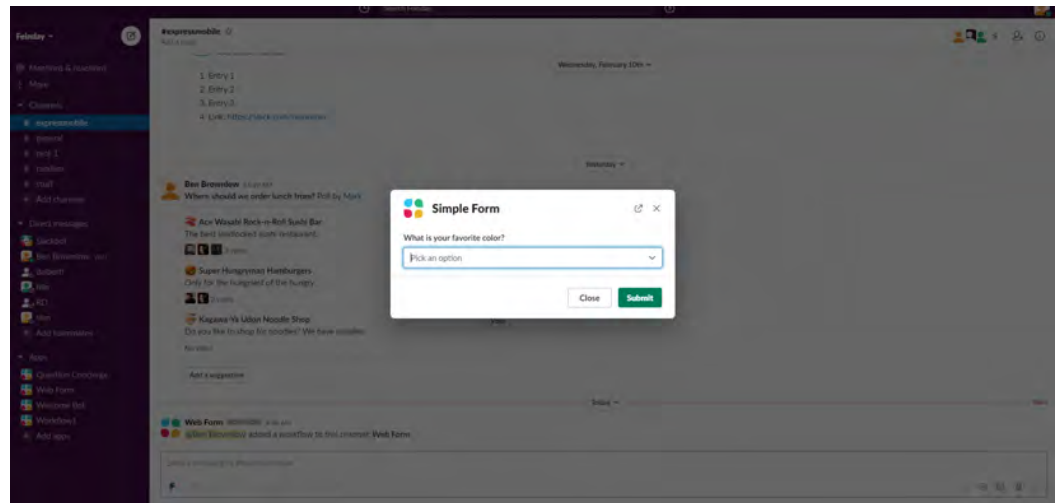
information stored in the database in order to generate and display at least a portion of one or more web pages. The player includes code that is device-dependent in order to allow the environment to work across a variety of devices such as personal computers (including laptops and desktops), tablets, and mobile phones.

112. The Accused Instrumentalities include a player or code that utilizes information stored in its database to generate for display one or more web pages as shown below. For example, as shown below, the Accused Instrumentalities allow a user to provide an input value and the corresponding input symbolic name to the web service through their device. The web service utilizes the user provided input value and the corresponding input symbolic name for generating one or more output values associated in the source code with an output symbolic name. The player receives the output symbolic name and corresponding one or more output values and provides instructions for a display of the device to present an output value in the defined UI object.

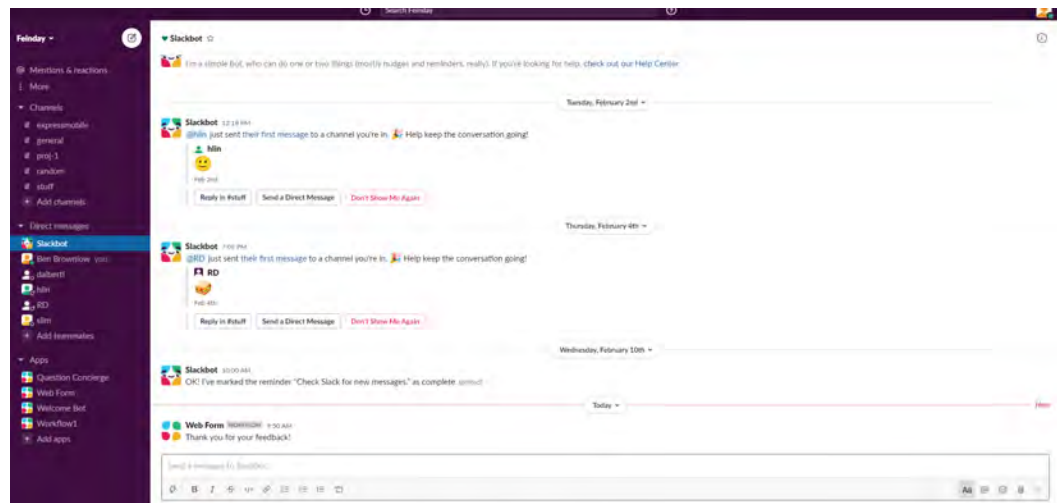
113. The player code on the device operates with the virtual machine to execute software instructions in order to receive the output symbolic name and output value. The instructions also provide for the display of this output value in the UI object in order to display the appropriate data to the user. This output value is presented in the UI object for display on the device to the user.











## View activity for your workflows

- 1 From your desktop, click your workspace name in the top left.
- 2 Select **Tools** from the menu, then click **Workflow Builder**.
- 3 From the **Your Workflows** tab, click a workflow to open it.
- 4 Click the **Activity** tab.
- 5 Browse the list of activity, or use the filters at the top to narrow results by status, date, or people.
- 6 Click **Show details** to see who triggered the workflow, and each step that was completed.



**Note:** You can only view workflow activity from the last 90 days.

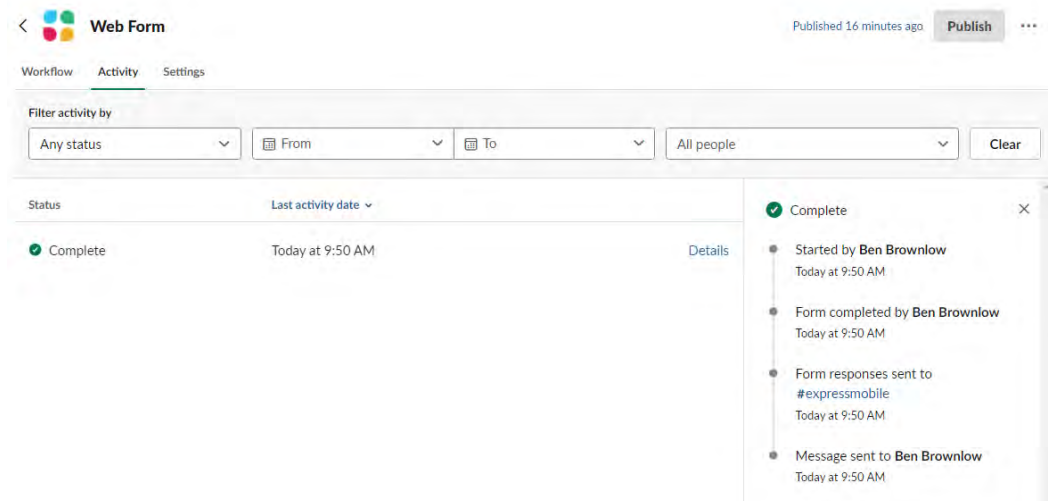
## View workflow activity in Workflow Builder

### Who can use this feature?

- Members who [collaborate on workflows](#)
- Standard, Plus, and Enterprise Grid plans

From the **Activity** tab in Workflow Builder, collaborators can see how people interact with their [workflows in Slack](#). View completed workflows, those that are in progress, and any workflow errors that may have occurred.

Source: <https://slack.com/help/articles/360055655493-View-workflow-activity-in-Workflow-Builder>



114. Slack was made aware of the '287 patent and its infringement thereof at least as early as December 19, 2019 when Express Mobile provided notice of Slack's infringement of the '287 patent to David Schellhase, General Counsel of Slack. Since at least the time Slack received notice, Slack has induced others to infringe at least one claim of the '287 patent under 35 U.S.C. § 271(b) by, among other things, and with specific intent or willful blindness, actively aiding and abetting others to infringe, including but not limited to Slack's clients, customers, and end users, whose use of the Accused Instrumentality constitutes direct infringement of at least one claim of the '287 patent. In particular, Slack's actions that aid and abet others such as

1 customers and end users to infringe include advertising and distributing the Accused  
 2 Instrumentality and providing instruction materials, training, and services regarding the Accused  
 3 Instrumentality. For example, Slack provides support services for the Accused Instrumentalities,  
 4 including providing instructions, guides, and online materials. *See, e.g.*, [slack.com/resources](https://slack.com/resources);  
 5 [slack.com/tutorials](https://slack.com/tutorials); [slack.com/features](https://slack.com/features); [slack.com/help](https://slack.com/help);  
 6 <https://slack.com/help/articles/360055655493-View-workflow-activity-in-Workflow-Builder>;  
 7 <https://api.slack.com/tutorials/design-expense-block-kit>[https://api.slack.com/tutorials/design-](https://api.slack.com/tutorials/design-expense-block-kit)  
 8 [expense-block-kit](https://api.slack.com/tutorials/design-expense-block-kit);  
 9 [https://www.workplace.com/help/work/1070177463385471?helpref=hc\\_global\\_nav](https://www.workplace.com/help/work/1070177463385471?helpref=hc_global_nav);  
 10 [https://www.workplace.com/help/work/300047317995067?helpref=about\\_content](https://www.workplace.com/help/work/300047317995067?helpref=about_content); and all  
 11 [related domains and subdomains](#). Slack has engaged in such actions with specific intent to cause  
 12 infringement or with willful blindness to the resulting infringement because Slack has had actual  
 13 knowledge of the '287 patent and knowledge that its acts were inducing infringement of the '287  
 14 patent since at least the date Slack received notice that such activities infringed the '287 patent.

15 115. Slack also contributes to infringement of the '287 patent by offering to sell, and/or  
 16 selling (including licensing) in the United States the Accused Instrumentalities, which are  
 17 especially made or adapted for use in an infringement of the '287 patent. The Accused  
 18 Instrumentalities are a material component for use in practicing the '287 patent and are not a  
 19 staple article of commerce suitable for substantial non-infringing use. For example, Slack's  
 20 platform, including Slack.com, the Slack Workflow Builder, the Slack mobile application, and  
 21 related software and servers constitute a material portion of the invention and has no substantial  
 22 non-infringing uses. Slack's clients, customers, and end users exercise control and obtain  
 23 beneficial use of the Accused Instrumentalities, as shown below. For example, as demonstrated  
 24 in connection with claim 1, Slack's clients, customers and end users exercise control over and  
 25 benefit from the use of Slack's Workflow Builder by using the authoring tool to define custom  
 26 UI objects for interacting with other Slack end users. Slack is thereby liable for contributory  
 27 infringement of the asserted claims of the '287 patent under 35 U.S.C. § 271(c).  
 28

116. Slack's infringement has damaged and continues to damage and injure Express Mobile.

**COUNT IV - INFRINGEMENT OF U.S. PATENT NO. 9,063,755**

117. Plaintiff incorporates by reference the allegations contained in paragraphs 1 to 116 above.

118. Slack has and is making, using, offering for sale, and/or selling (including licensing) in the United States Slack's platform, including Slack.com, Slack Workflow Builder, the Slack mobile application available for Apple iOS devices (including Apple TV, iPads, iPhones and iPod Touch) on Apple's App Store, Android devices (including phones and tablets) on Google Play, and related software and servers, that afford users access to a system for generating code to provide content on a display of a device (collectively, the "Accused Instrumentalities") which infringes, either literally or under the doctrine of equivalents, one or more claims of the '755 patent in violation of 35 U.S.C. § 271(a).

119. Upon information and belief, Slack has directly infringed at least claim 1 of the '755 patent through the Accused Instrumentalities that comprise a system for generating code to provide content on the display of a device for each of its users generate code to provide content on a display of a device

120. The Accused Instrumentalities comprise a system for generating code to provide content on the display of a device for each of its users.

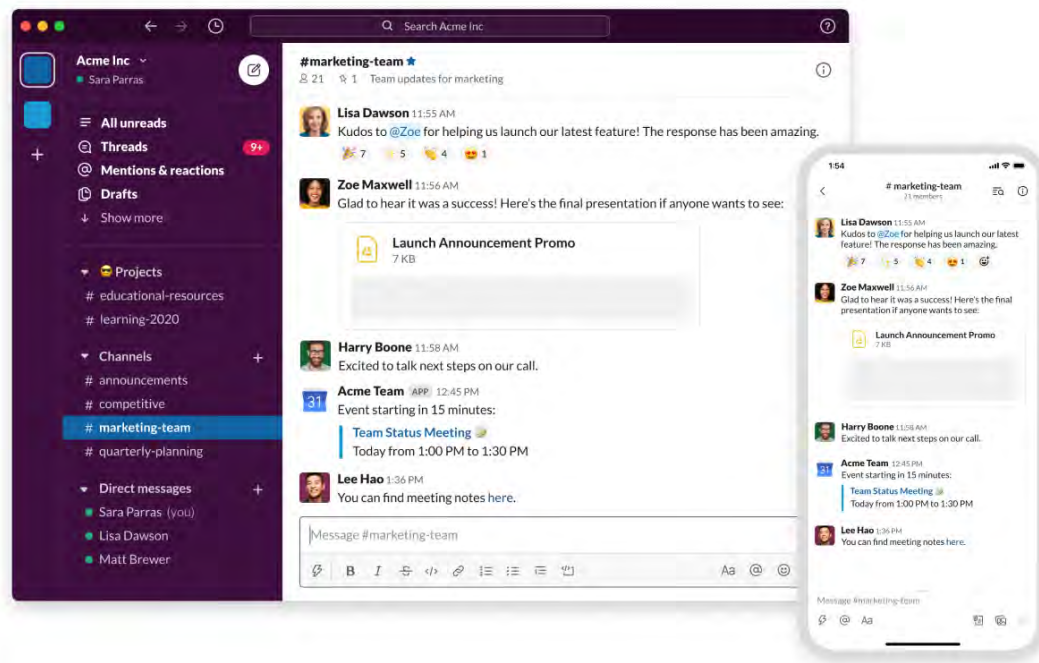
SLACK FEATURES

## One platform for your team and your work

All the features of Slack work together so you can, too.

GET STARTED

SEE ALL FEATURES



Source: <https://slack.com/features>

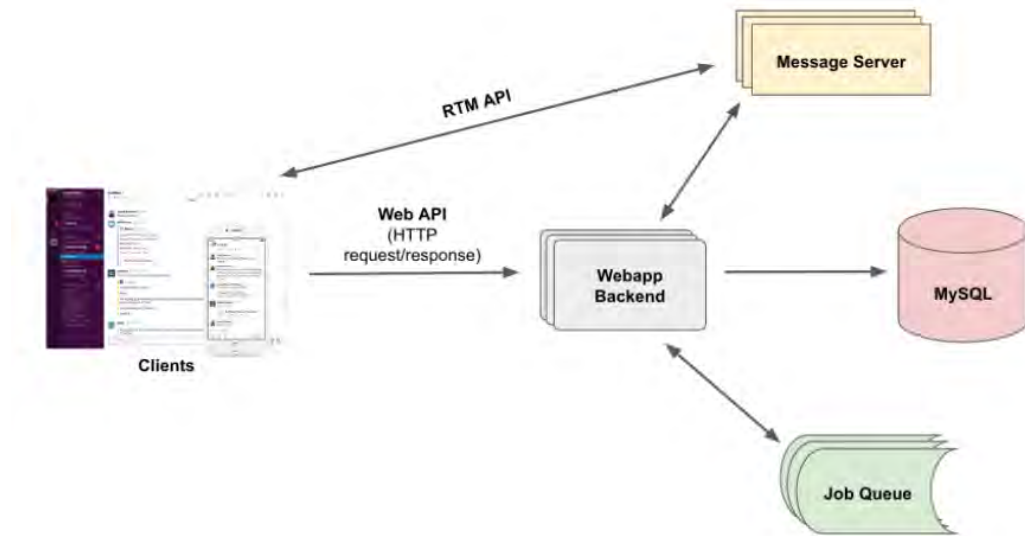
121. The Accused Instrumentalities include computer memory to store a registry of user selections and other information relating to a user's web site, web pages, and the web components or elements that are intended to be displayed.

Before we dive into shared channels, let's take a look at how Slack works at a high level. Slack implements a client-server architecture where clients (mobile, desktop, web and apps) talk to two backend systems:

- 1 webapp servers, which handle HTTP request/response cycles and communicate with other systems like the main databases and job queues
- 2 real-time message servers, which fan out messages, profile changes, user presence updates, and a host of other events to clients

For example, a new message posted in a channel is sent via an API call to the webapp servers, where it is sent to the message servers and is persisted in the database. The real-time message servers receive these new messages and send them to connected clients over web sockets, based on who is in the channel.

Source: <https://slack.engineering/how-slack-built-shared-channels/#:~:text=%20Slack%20implements%20a%20client-server%20architecture%20where%20clients,a%20host%20of%20other%20events%20to...%20More%20>



*Basic Slack architecture*

Source: <https://slack.engineering/how-slack-built-shared-channels/#:~:text=%20Slack%20implements%20a%20client-server%20architecture%20where%20clients,a%20host%20of%20other%20events%20to...%20More%20>


122. The Accused Instrumentalities includes symbolic names required for evoking one or more web components each related to a set of inputs and outputs of a web service obtainable over a network. The symbolic names, such as parameter names, are character strings that do not contain either a persistent address or pointer to an output value accessible to the web service.


123. The Accused Instrumentalities include a feature called Workflow Builder, which offers a set of tools to help users automate processes into workflows (or custom applications) in their Slack workspace. Workflows require the use of a symbolic name, or parameter value, to evoke a web component related to a set of inputs and outputs of a web service obtainable over a network.



# Guide to Workflow Builder

## Who can use this feature?

 By default, all **members** of a workspace can [create workflows](#). Any **member** or **guest** can use workflows added to channels they belong to.

 **Standard, Plus, and Enterprise Grid** plans

[Workflow Builder](#) offers a set of tools — no coding required! — to help you automate routine processes into workflows in your Slack workspace. Workflows in Slack can be as simple or as complex as you'd like, and can even be connected to other apps and services you use to get work done.

Read on to learn about Workflow Builder, then use our tutorials as a guide to build your workflow.

Source: <https://slack.com/help/articles/360035692513-Guide-to-Workflow-Builder>

## Modals

Modals are dedicated windows that allow your app to collect input from users and display dynamic information in a focused space. Modals must be initiated by a user (with features like buttons, slash commands, and other invocative actions).

Source: <https://api.slack.com/tutorials/design-expense-block-kit>

## Workflow Builder glossary

<b>Trigger</b>	How your workflow will start. You can choose a ⚡ <b>shortcuts menu</b> trigger to launch your workflow from the menu in a channel, or another trigger to start your workflow automatically when a certain action is taken.
<b>Step</b>	How people interact with your workflow. You can add steps that send messages or custom forms, or use <a href="#">steps from Slack apps</a> to link your workflow to other tools.
<b>Variable</b>	Information submitted to your workflow that can be referenced in steps. More variables will be available to choose from as you add steps to your workflow.
<b>Collaborator</b>	A member of your workspace with permission to manage your workflow. Any collaborator can change or remove steps, manage other collaborators, and unpublish or delete a workflow.






Source: <https://slack.com/help/articles/360035692513-Guide-to-Workflow-Builder>

## Set up your workflow

While you'll first be presented with picking a name for your workflow, the more important decision you'll need to make is how your workflow will start.

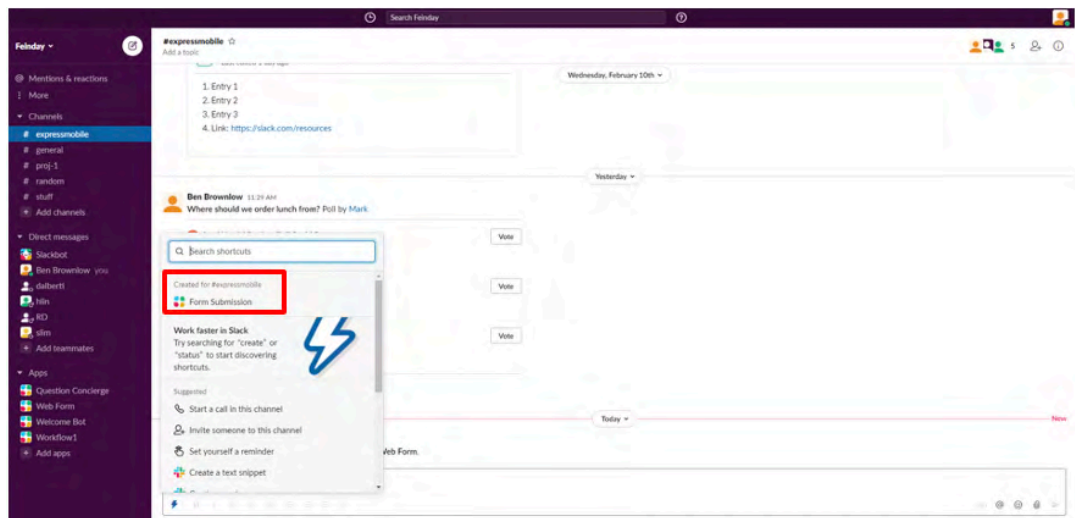
### Choose a trigger

The trigger you pick for your workflow determines how it will start. Workflows are channel-specific, and can only be used in the channel you choose.

Trigger	How it works
 Shortcuts menu	Your workflow starts when someone selects it from the shortcuts menu.
 New channel member	Your workflow starts automatically when someone joins the channel.
 Emoji reaction	Your workflow starts automatically when someone in the channel adds a certain <a href="#">emoji reaction</a> to a message.
 Scheduled date & time	Your workflow starts automatically at a set date and time.
 Webhook*	Your workflow starts automatically <a href="#">when a custom webhook sends a web request to Slack</a> from another app or service.

\*Creating workflows with webhooks requires some technical expertise.

Source: <https://slack.com/help/articles/360053571454-Set-up-a-workflow-in-Slack>



124. The Accused Instrumentalities include an address of the web service, as shown below.

## Network settings

Slack must have a persistent connection between our messaging server and members' apps or browsers. To do so, Slack uses WebSockets over port 443.

Proxies and firewalls can sometimes interrupt this connection. If someone using Slack can't connect from a specific location, or if there's a WebSocket failure in our [Slack connection test](#), you'll need to adjust your proxy or firewall to keep the connection to Slack open. Here's how:

- ✔ Visit <https://my.slack.com/help/urls> and add all specified URLs to your allowlist. If your environment requires access to multiple workspaces and/or orgs, visit that page from each workspace and/or org.
- ✔ Check if your proxy is running SSL decryption. If it is, the proxy must either support WebSockets, or you'll need to exempt the following domains: \*.lb.slack-msgs.com wss-primary.slack.com wss-backup.slack.com wss-mobile.slack.com

Network administrators can restrict access to specific Slack workspaces using [proxy header injection](#) on their network. The header injection solution has the same network requirements as above.

```

1      "*.chime.aws",
2      "*.lb.slack-msgs.com",
3      "a.slack-edge.com",
4      "a.slack-imgs.com",
5      "admin.slack.com",
6      "alpha.slack.com",
7      "api.slack.com",
8      "app.slack.com",
9      "assets.slack.com",
10     "auth.slack.com",
11     "avatars.slack-edge.com",
12     "b.slack-edge.com",
13     "b.slack-imgs.com",
14     "beta.slack.com",
15     "blog.slack.com",
16     "ca.slack-edge.com",
17     "downloads.slack-edge.com",
18     "edgeapi.slack.com",
19     "email.slack.com",
20     "email2.slack.com",
21     "email3.slack.com",
22     "email4.slack.com",
23     "emoji.slack-edge.com",
24     "example.slack.com",
25     "feinday.slack.com",
26     "files-edge.slack.com",
27     "files-origin.slack.com",
28     "files.slack.com",
29     "global-upload-edge.slack.com",
30     "go-beta.slack.com",
31     "go-debug.slack.com",
32     "go.slack.com",
33     "help.slack.com",
34     "hooks.slack.com",
35     "join.slack.com",
36     "my.slack.com",
37     "oauth2.slack.com",
38     "platform-tls-client.slack.com",
39     "platform.slack-edge.com",
40     "slack-email.slack.com",
41     "slack-files.com",
42     "slack-imgs.com",
43     "slack-infra.slack.com",
44     "slack-redirect.com",
45     "slack-redirect.net",
46     "slack.com",
47     "slack.global.ssl.fastly.net",
48     "slackb.com",
49     "spellcheck.slack-edge.com",
50     "status.slack.com",
51     "try.slack.com",
52     "universal-upload-edge.slack.com",
53     "upload.slack.com",
54     "wss-backup.slack.com",
55     "wss-mobile.slack.com",
56     "wss-primary.slack.com"

```

Source: <https://slack.com/help/articles/360001603387-Manage-Slack-connection-issues>

125. The Accused Instrumentalities include an authoring tool configured to define a UI object for presentation on the display. The UI object corresponds to a web component which consists of an input of the web service and an output of the web service. For example, Workflow

Builder allows a user to define a UI object corresponding to a web component for presentation on the display.

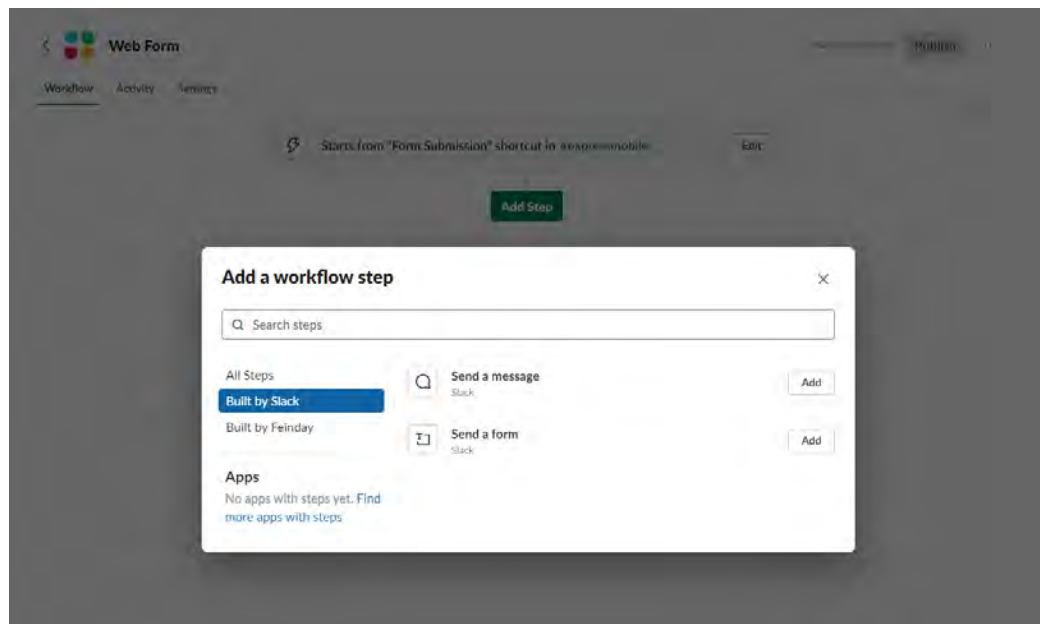
### Form steps

Add **forms** to your workflow to organize and collect information. Forms can include space for custom responses, or prompt people to choose options from a list. Once a form has been completed, you can choose to send responses to a channel or a specific person via DM.



**Note:** Form responses cannot be edited once they've been submitted via Workflow Builder.

Source: <https://slack.com/help/articles/360053571474-Add-steps-to-a-workflow>





**Web Form**

**Send a form**

Title: Simple Form

Questions

Add a question: What is your favorite color?

Choose a question type: Select from a list (selected)

Short answer

Long answer

Select a person

Select a channel or DM

Default selection: No selection

☒ Make this required

Add Question

☒ Send submitted responses to a channel or to someone in a DM

Cancel Save

**Web Form**

**Send a form**

Questions

Add a question: What is your favorite color?

Choose a question type: Select from a list

List Items:

Red	↑	↓	✕
Green	↑	↓	✕
Blue	↑	↓	✕

Add List Item

Default selection: No selection

☒ Make this required

Add Question

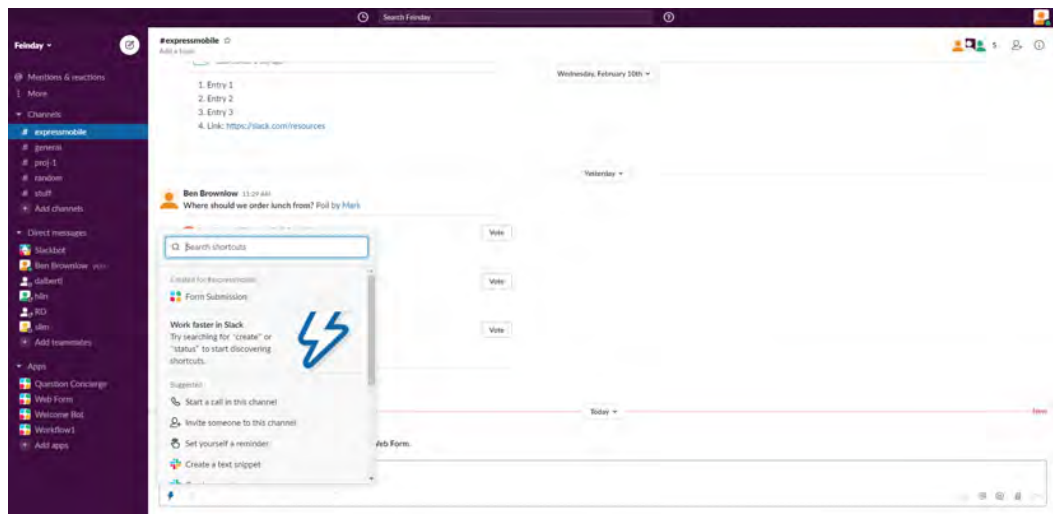
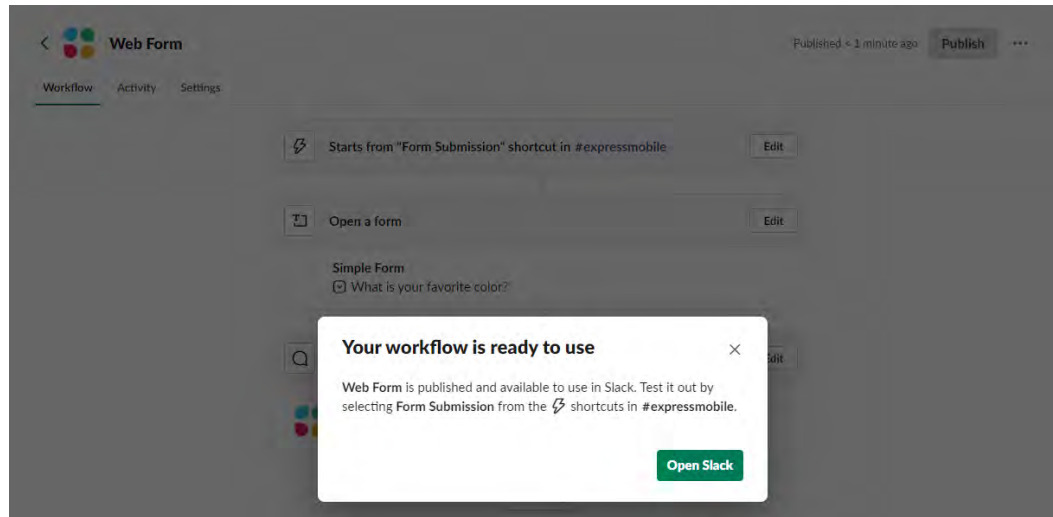
☒ Send submitted responses to a channel or to someone in a DM

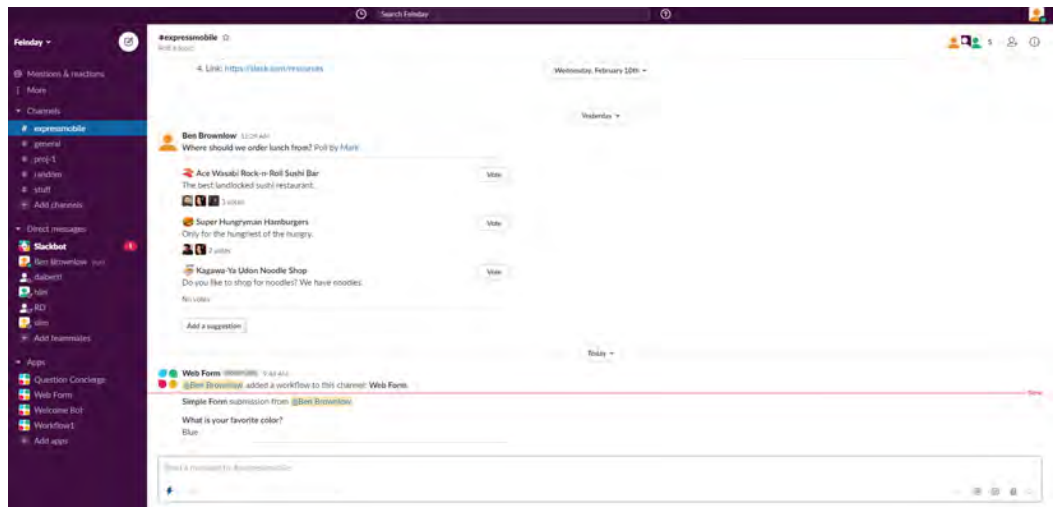
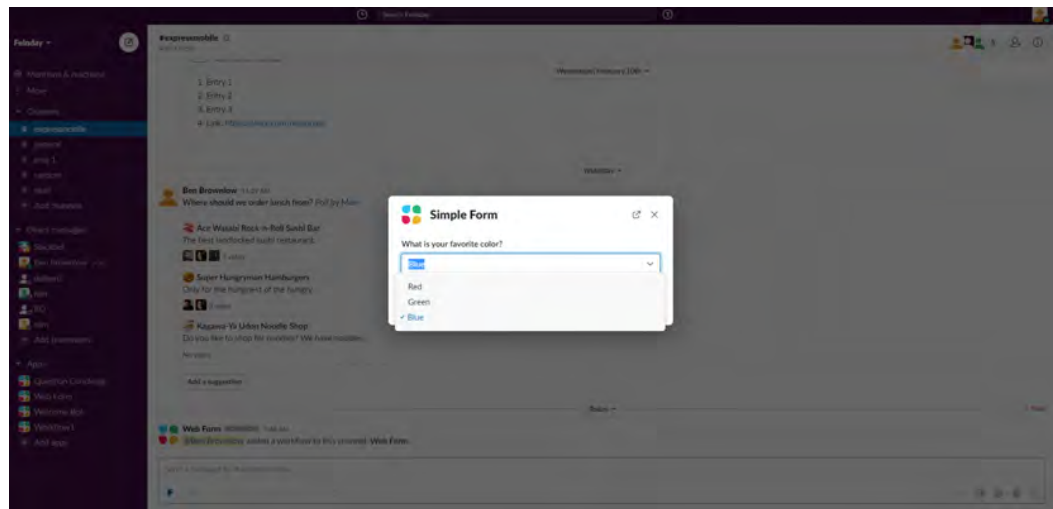
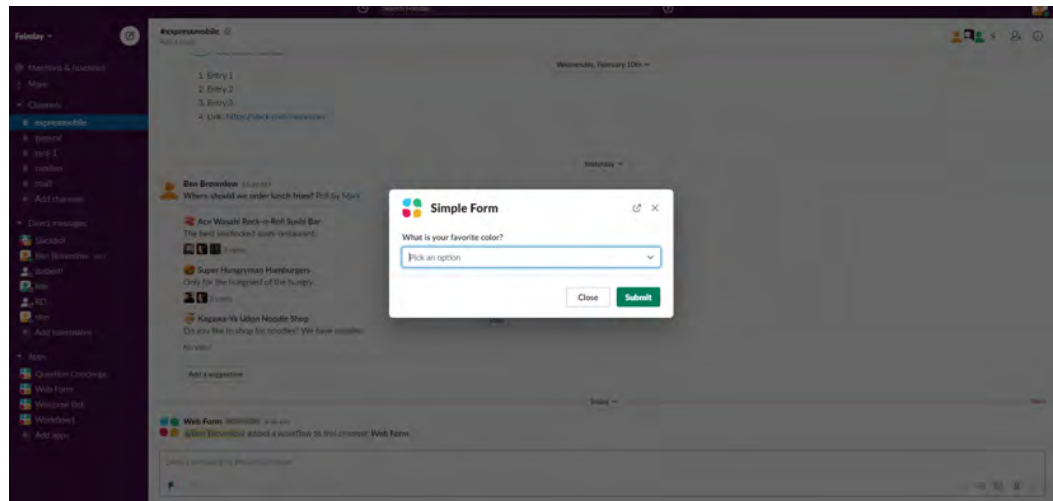
# expressmobile

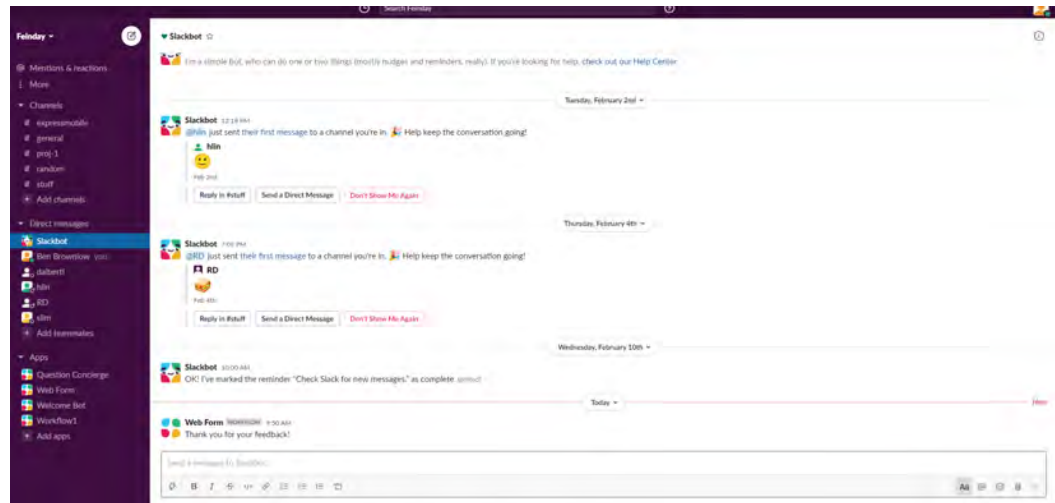
Cancel Save

126. When a user of the Accused Instrumentalities uses the authoring tool to create a Workflow, the authoring tool accesses the computer memory to select the associated UI objects

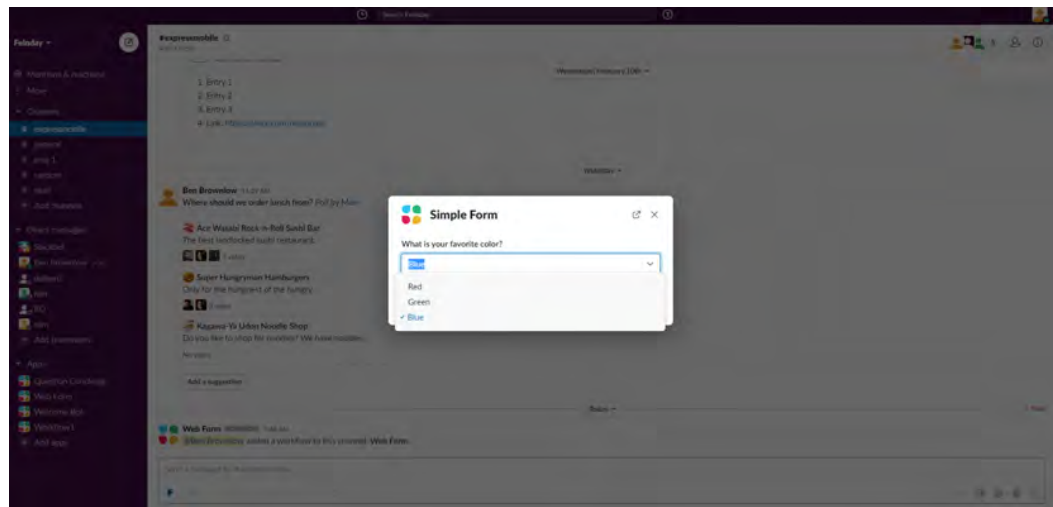
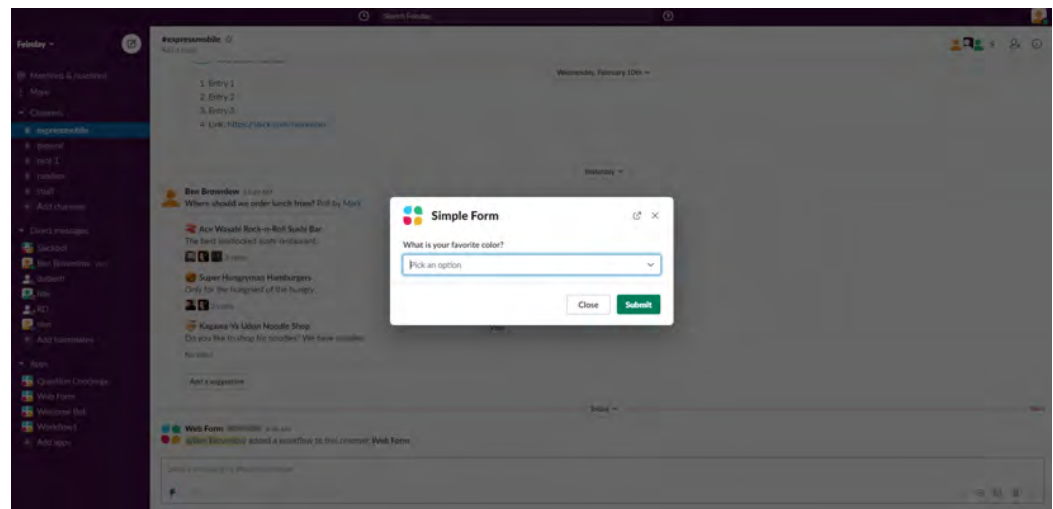
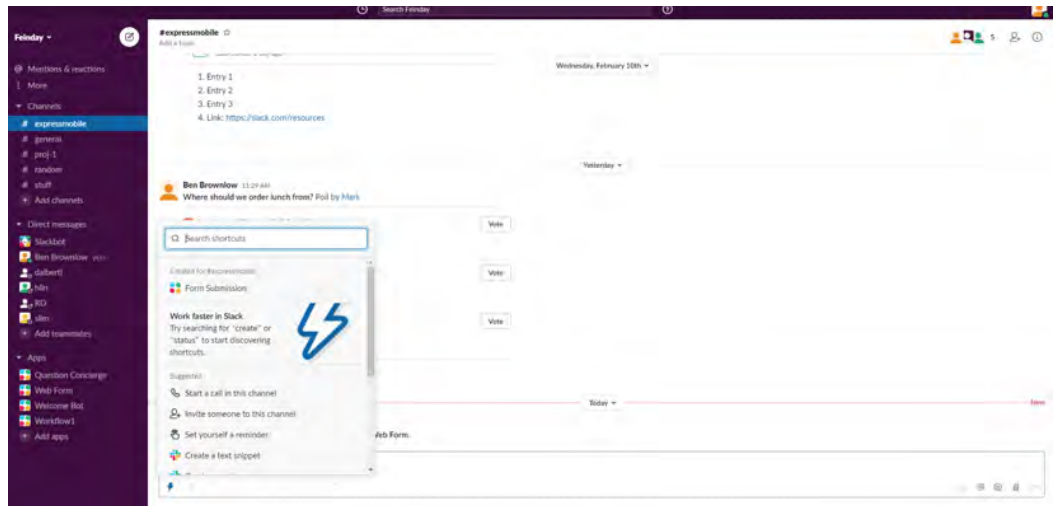
to allow the user to do so. Once selected, Slack associates the symbolic name, such as a parameter name, with the selected defined UI object. The symbolic name is only available to corresponding UI objects.

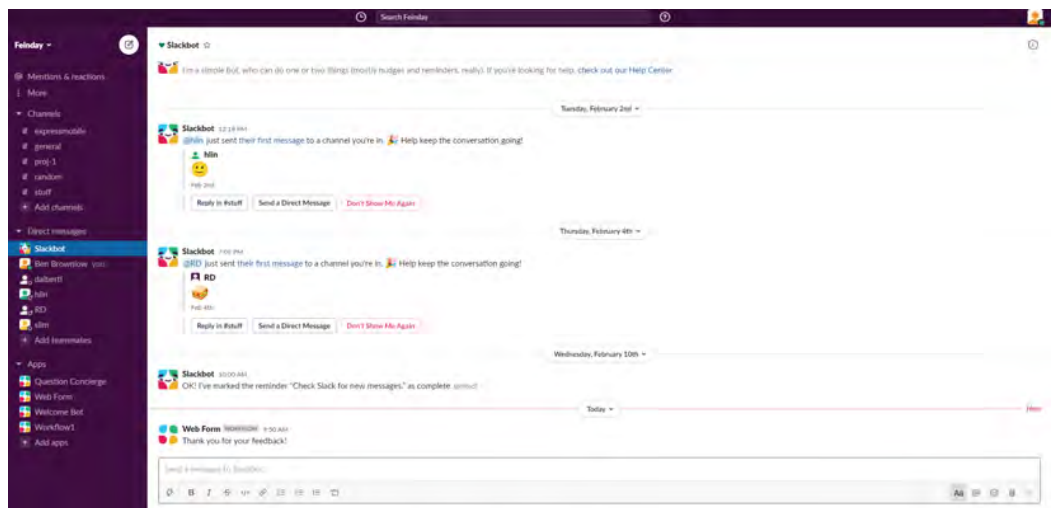
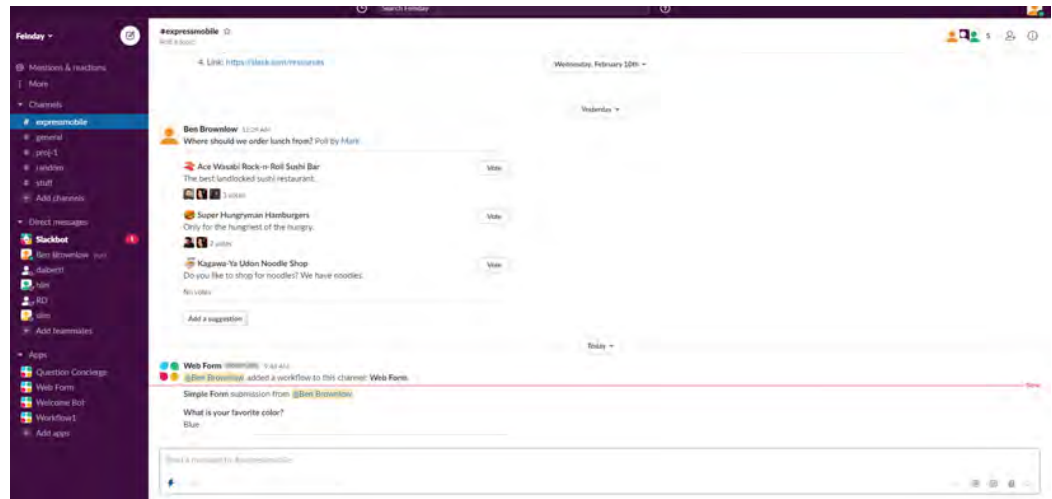






127. The application is provided, for example, in the form of software files and associated data for the web page view(s) that are stored in the Slack database. When a browser is used to access Slack, it uses a player which interacts with the application and data stored on the Slack server. The player is, for example, software provided to the device in connection with the application. The player accesses and renders the data to generate the web page viewed by the user. The player operates with the virtual machine (for example, Microsoft Internet Explorer uses the Trident or Blink virtual machine, depending on the software version) and the information stored in the database in order to generate and display at least a portion of one or more web pages. The player includes code that is device-dependent in order to allow the environment to work across a variety of devices such as personal computers (including laptops and desktops), tablets, and mobile phones.





128. The Accused Instrumentalities provide the application and the player to the device to be executed on the device. When a user of the device provides one or more input values associated with an input symbolic name to an input of the defined UI object, the device provides the user provided one or more input values and corresponding input symbolic name to the web service. The web service utilizes the input symbolic name and the user provided one or more input values for generating one or more output values having an associated output symbolic name.



`actions.value`

Set by your app when you composed the blocks, this is the `value` that was specified in the interactive component when an interaction happened. For example, a `select` menu will have multiple possible values depending on what the user picks from the menu, and `value` will identify the chosen option. See the reference guide for the interactive element you're using for more info on the `value` field.

Source: <https://api.slack.com/reference/interaction-payloads/block-actions>

## Handling `block_actions` payloads

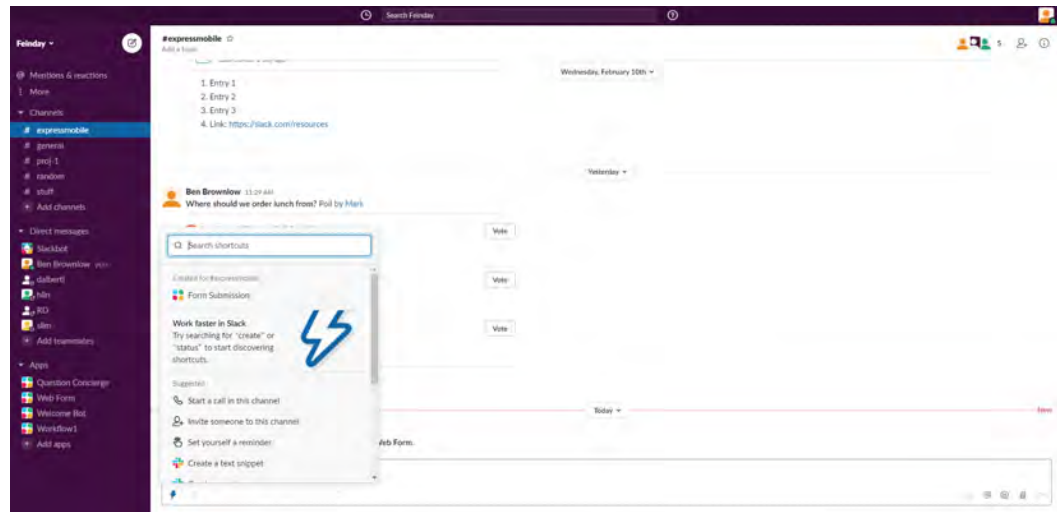
When someone uses an interactive component in your app's modal views, the app will receive a `block_actions` payload.

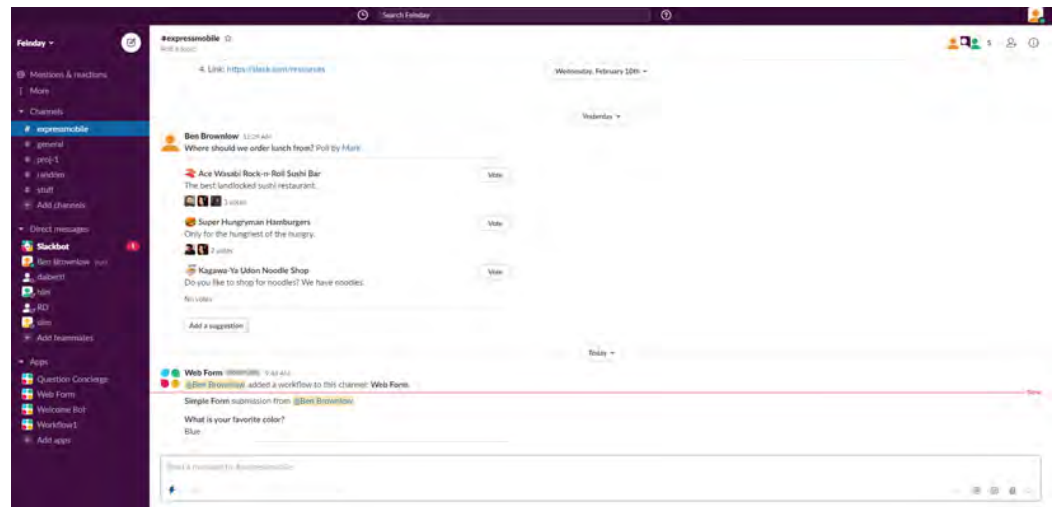
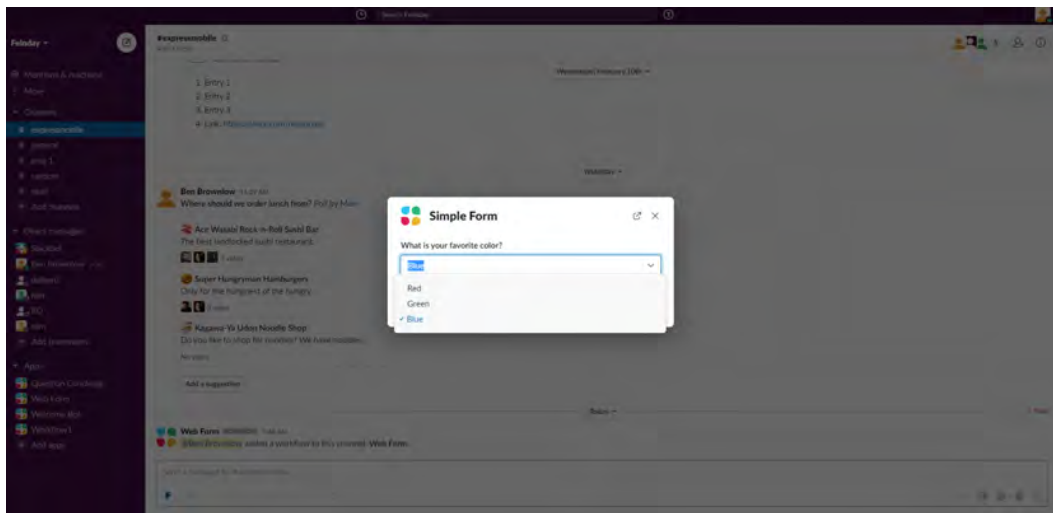
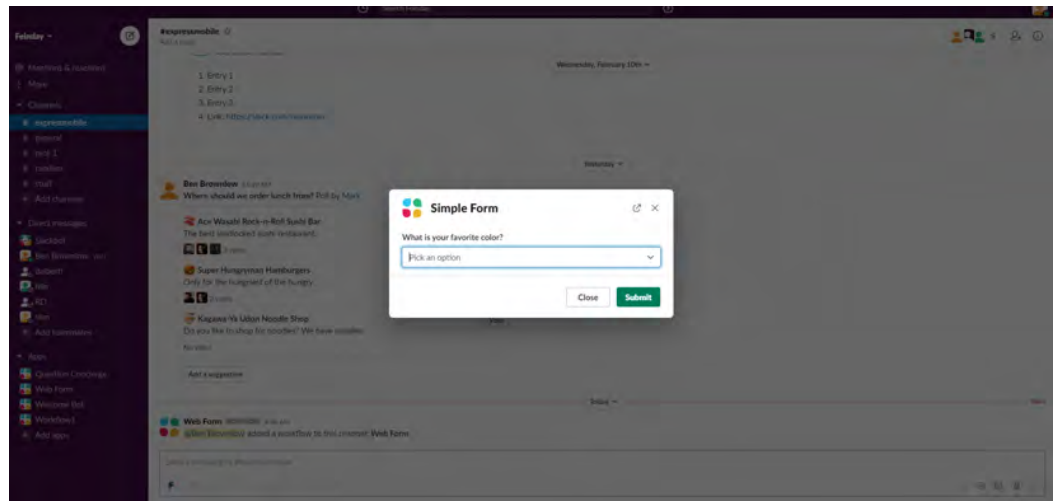
This **does not** apply to components included in an `input` block (see below for details about those).

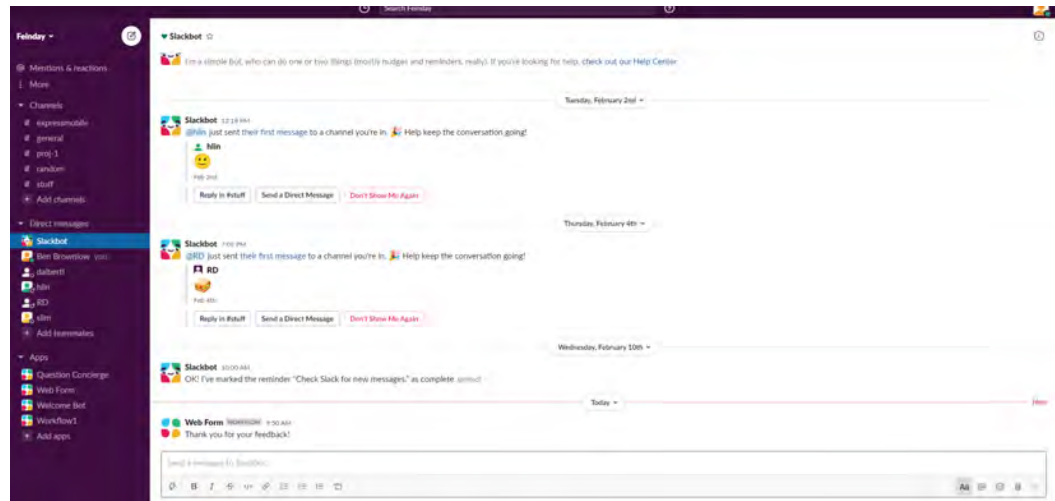
Once processed, the information in the `block_actions` payload can be used to respond to the interaction, which we'll explain later.

Source: [https://api.slack.com/surfaces/modals/using#response\\_actions](https://api.slack.com/surfaces/modals/using#response_actions)

129. The player receives the output symbolic name and corresponding one or more output values and provides instructions for the display of the device to present an output value in the defined UI object.







130. The Accused Instrumentalities also include an application that allows users to use Slack on mobile devices.

## Minimum requirements for using Slack

If you're curious about the technical requirements needed to run Slack, read on to learn which operating systems (OS) and browsers we support.

### Desktop apps

Operating System	Requirements
<a href="#">MacOS</a>	OS X 10.10 or above
<a href="#">Windows</a>	Windows 7 or above
<a href="#">Linux</a>	Fedora 28 or above Ubuntu LTS releases 16.04 or above Red Hat Enterprise Linux 7.0 or above

## Mobile apps

Operating System	Requirements
<a href="#">iOS</a>	iOS 12.2 or above
<a href="#">Android</a>	Android 6.0 or above

## Web browsers

Browser	Requirements
Chrome	Version 66 or above
Firefox	Version 60 or above
Safari	Version 11.0 or above
Microsoft Edge	Version 41 or above



**Note:** Google Chrome is the only browser that supports [Slack calls](#).

## Unsupported browsers and operating systems

To deliver the best possible Slack experience, we keep our list of supported browsers and operating systems short. We don't fix bugs or issues for unsupported browsers, so we do our best to block them from accessing Slack.

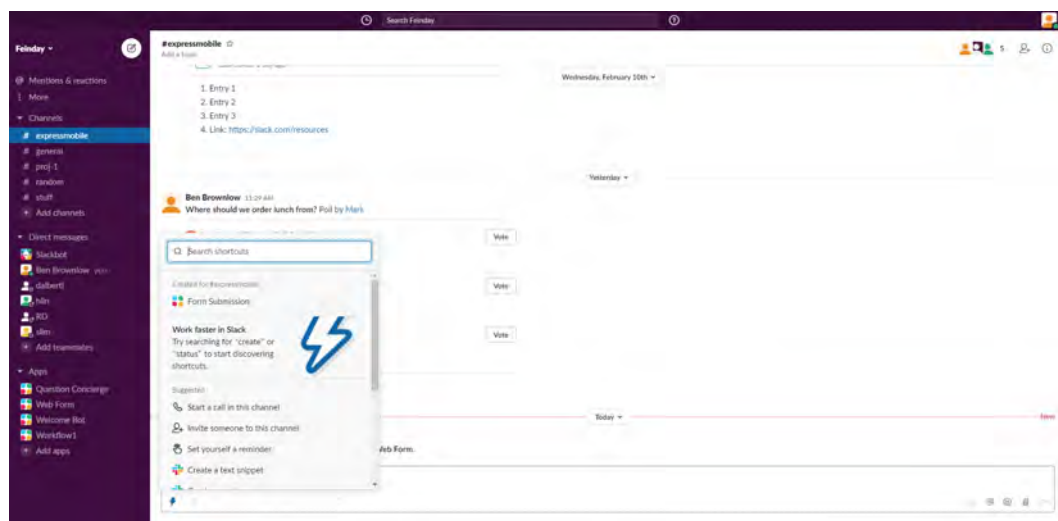
Source:: <https://slack.com/help/articles/115002037526-Minimum-requirements-for-using-Slack>

131. The application is provided, for example, in the form of software files and associated data for the web page view(s) that are stored in the Slack database. When a browser is used to access Slack, it uses a player which interacts with the application and data stored on the Slack server. The player is, for example, software provided to the device in connection with the application. The player accesses and renders the data to generate the web page viewed by the user. The player operates with the virtual machine (for example, Microsoft Internet Explorer uses the Trident or Blink virtual machine, depending on the software version) and the

information stored in the database in order to generate and display at least a portion of one or more web pages. The player includes code that is device-dependent in order to allow the environment to work across a variety of devices such as personal computers (including laptops and desktops), tablets, and mobile phones.

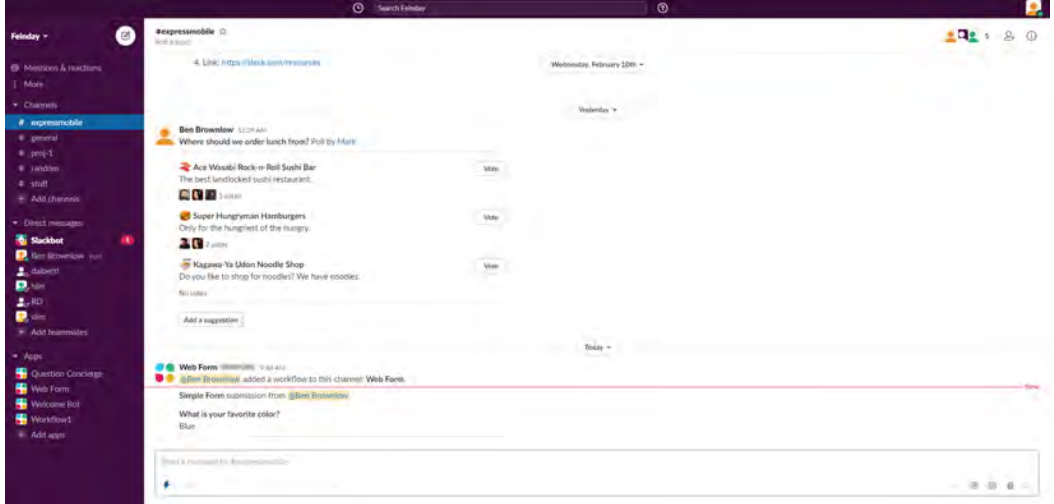
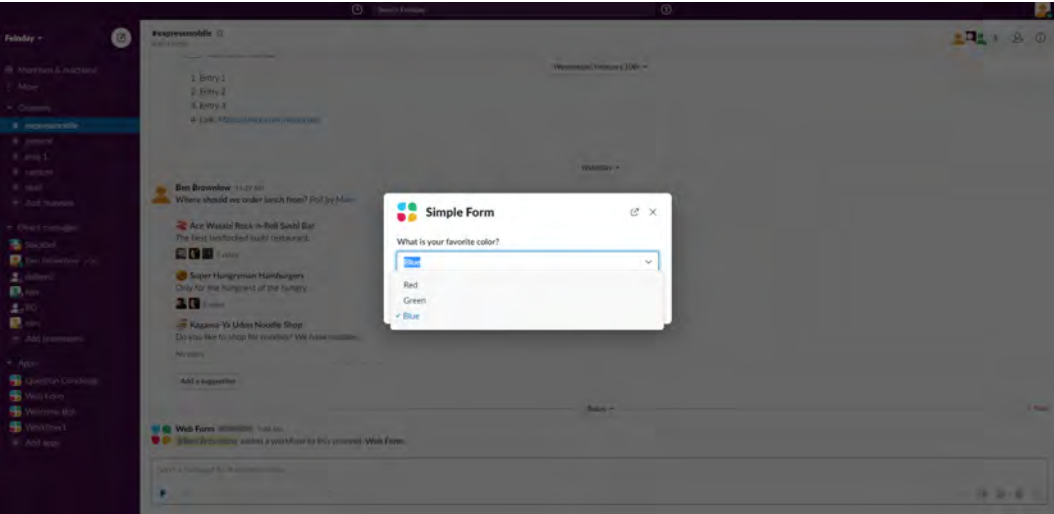
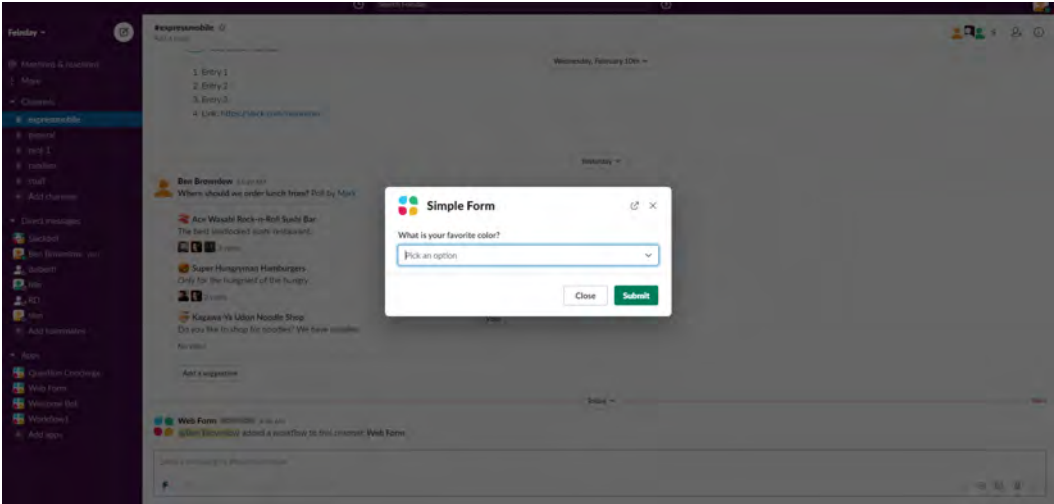
132. The Accused Instrumentalities include a player or code that utilizes information stored in its database to generate for display one or more web pages as shown below. For example, as shown below, the Accused Instrumentalities allow a user to provide an input value and the corresponding input symbolic name to the web service through their device. The web service utilizes the user provided input value and the corresponding input symbolic name for generating one or more output values associated in the source code with an output symbolic name. The player receives the output symbolic name and corresponding one or more output values and provides instructions for a display of the device to present an output value in the defined UI object.

133. The player code on the device operates with the virtual machine to execute software instructions in order to receive the output symbolic name and output value. The instructions also provide for the display of this output value in the UI object in order to display the appropriate data to the user. This output value is presented in the UI object for display on the device to the user.

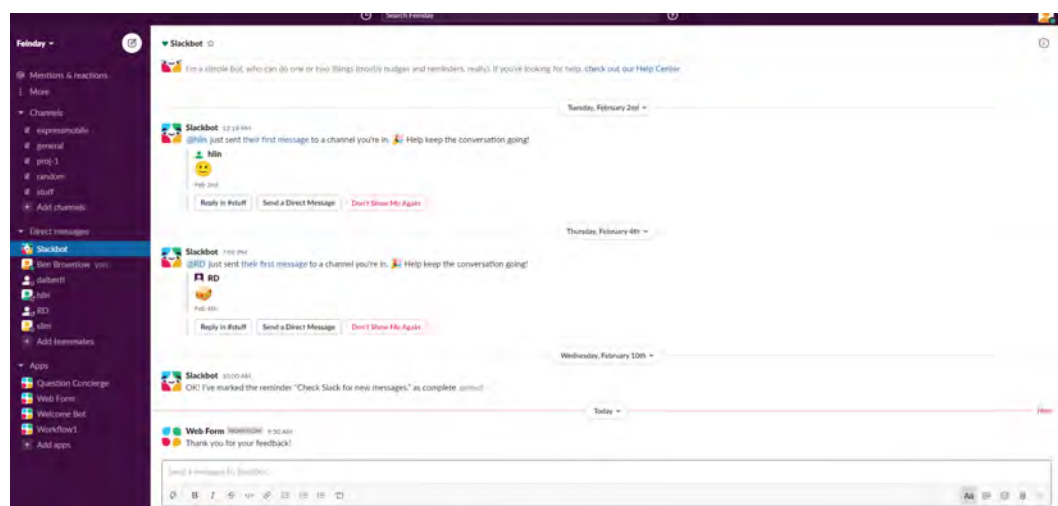




1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28







## View activity for your workflows



- 1 From your desktop, click your workspace name in the top left.
- 2 Select **Tools** from the menu, then click **Workflow Builder**.
- 3 From the **Your Workflows** tab, click a workflow to open it.
- 4 Click the **Activity** tab.
- 5 Browse the list of activity, or use the filters at the top to narrow results by status, date, or people.
- 6 Click **Show details** to see who triggered the workflow, and each step that was completed.



**Note:** You can only view workflow activity from the last 90 days.

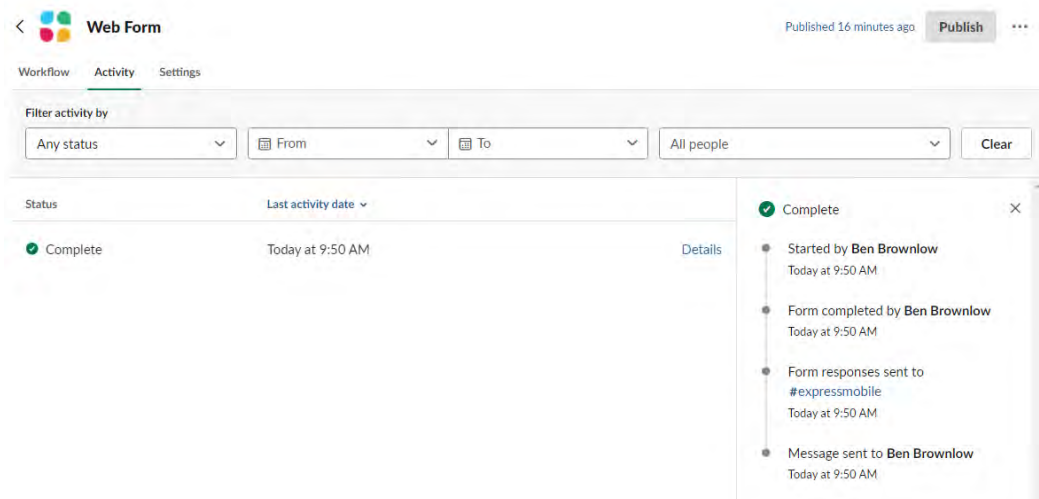
## View workflow activity in Workflow Builder

### Who can use this feature?

 **Members** who [collaborate on workflows](#)  
 **Standard, Plus, and Enterprise Grid** plans

From the **Activity** tab in Workflow Builder, collaborators can see how people interact with their [workflows in Slack](#). View completed workflows, those that are in progress, and any workflow errors that may have occurred.

Source: <https://slack.com/help/articles/360055655493-View-workflow-activity-in-Workflow-Builder>



134. Slack was made aware of the '755 patent and its infringement thereof at least as early as December 19, 2019 when Express Mobile provided notice of Slack's infringement of the '755 patent to David Schellhase, General Counsel of Slack. Since at least the time Slack received notice, Slack has induced others to infringe at least one claim of the '755 patent under 35 U.S.C. § 271(b) by, among other things, and with specific intent or willful blindness, actively aiding and abetting others to infringe, including but not limited to Slack's clients, customers, and end users, whose use of the Accused Instrumentality constitutes direct infringement of at least

one claim of the '755 patent. In particular, Slack's actions that aid and abet others such as customers and end users to infringe include advertising and distributing the Accused Instrumentality and providing instruction materials, training, and services regarding the Accused Instrumentality. For example, Slack provides support services for the Accused Instrumentalities, including providing instructions, guides, and online materials. *See, e.g.*, slack.com/resources; slack.com/tutorials; slack.com/features; slack.com/help; <https://slack.com/help/articles/360055655493-View-workflow-activity-in-Workflow-Builder>; <https://api.slack.com/tutorials/design-expense-block-kit>; <https://api.slack.com/tutorials/design-expense-block-kit>; [https://www.workplace.com/help/work/1070177463385471?helpref=hc\\_global\\_nav](https://www.workplace.com/help/work/1070177463385471?helpref=hc_global_nav); [https://www.workplace.com/help/work/300047317995067?helpref=about\\_content](https://www.workplace.com/help/work/300047317995067?helpref=about_content); and all [related domains and subdomains](#). Slack's clients, customers, and end users exercise control and obtain beneficial use of the Accused Instrumentalities. For example, as demonstrated in connection with claim 1, Slack clients, customers, and end users exercise control over and benefit from the use of Slack's Workflow Builder by using the authoring tool to define custom UI objects for interacting with other Slack end users. Slack is thereby liable for induced infringement of the asserted claims of the '755 patent under 35 U.S.C. § 271(b).

135. Slack also contributes to infringement of the '755 patent by offering to sell, and/or selling (including licensing) in the United States the Accused Instrumentalities, which are especially made or adapted for use in an infringement of the '755 patent. The Accused Instrumentalities are a material component for use in practicing the '755 patent and are not a staple article of commerce suitable for substantial non-infringing use. For example, Slack's platform, including Slack.com, the Slack Workflow Builder, the Slack mobile application, and related software and servers constitute a material portion of the invention and has no substantial non-infringing uses. Slack's clients, customers, and end users exercise control and obtain beneficial use of the Accused Instrumentalities, as shown below. For example, as demonstrated in connection with claim 1, Slack's clients, customers and end users exercise control over and benefit from the use of Slack's Workflow Builder by using the authoring tool to define custom

1 UI objects for interacting with other Slack end users. Slack is thereby liable for contributory  
2 infringement of the asserted claims of the '755 patent under 35 U.S.C. § 271(c).

3 136. Slack's infringement has damaged and continues to damage and injure Express  
4 Mobile.

5 **PRAYER FOR RELIEF**

6 WHEREFORE, Plaintiff requests that the Court enter judgment for Plaintiff and against  
7 Defendant as follows:

8 a. That U.S. Patent No. 6,546,397 be judged valid, enforceable, and infringed by  
9 Defendant;

10 b. That U.S. Patent No. 9,928,044 be judged valid, enforceable, and infringed by  
11 Defendant;

12 c. That U.S. Patent No. 9,471,287 be judged valid, enforceable, and infringed by  
13 Defendant;

14 d. That U.S. Patent No. 9,063,755 be judged valid, enforceable, and infringed by  
15 Defendant;

16 e. That Plaintiff be awarded judgment against Defendant for damages together with  
17 interests and costs fixed by the Court including an accounting of all infringements and/or damages  
18 not presented at trial;

19 f. That the Court declare this an exceptional case and award Plaintiff its attorneys'  
20 fees, as provided by 35 U.S.C. § 285; and

21 g. That Plaintiff be awarded such other and further relief as this Court may deem just  
22 and proper.

**JURY DEMAND**

Plaintiff respectfully requests a jury trial on all issues so triable.

Dated: March 23, 2021

Respectfully submitted,

By: /s/ Robert Kramer  
Robert Kramer

FEINBERG DAY KRAMER ALBERTI  
LIM TONKOVICH & BELLOLI LLP

*Attorneys for* Express Mobile, Inc.